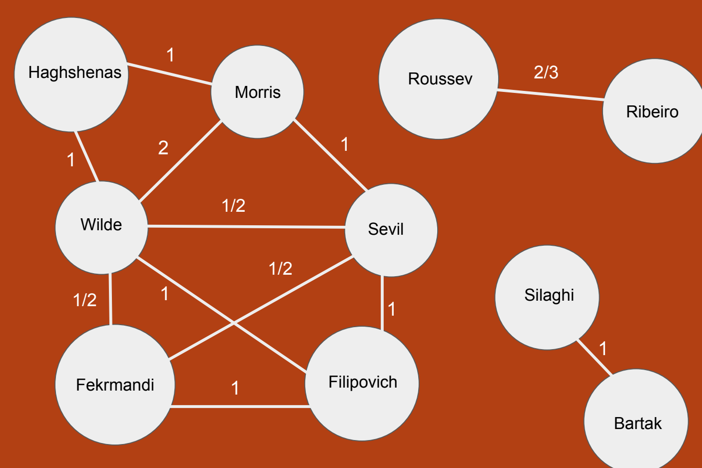
FCRAR 2020 - Proceedings

33rd Florida Conference on Recent Advances in Robotics

FCRAR 2020 - Proceedings

33rd Florida Conference on Recent Advances in Robotics



May 15th 2020

Florida Institute of Technology
High Tech with a Human Touch™

Edited by: M. C. Silaghi

Melbourne, Florida, USA
May 15th 2020

Cover by: Ioana Silaghi

PROCEEDINGS

Florida Conference on Recent Advances in Robotics (FCRAR)

Hosted by the

Florida Institute of Technology

May 15, 2020

Organizing Committee Members:

Robotics is an important research area at the intersection of most engineering fields and where recent advances impact on the whole society.

Independent Publisher

Any remaining samples from this edition can be obtained from:
Marius C. Silaghi
Computer Engineering and Sciences Department
Florida Institute of Technology
150 West University Boulevard
Melbourne FL-32901
USA

Email: msilaghi@fit.edu

Open Peer-Review for FCRAR 2020

In 2020 the 33rd Florida Conference on Recent Advances in Robotics (FCRAR) was held solely online in the context of the international Covid-19 lockdown. The conference experienced 22 article submissions, of which 9 were voluntarily withdrawn by their authors subsequent to the review cycle.

Presentations of articles were made during Zoom sessions featuring an average of 23 participants, with peaks of up to 27 participants. Each presentation was allocated 25 minutes, followed by 5 minutes of questions. The presentations were grouped into sessions of 3 articles, with breaks of 15 minutes between sessions, with sessions on *Manufacturing*, *Planning* and *Vision*. Recordings of presentations are uploaded in the FCRAR2020 YouTube channel, and are linked from the main conference website at `fcrar2020.fit.edu`.

Traditionally FCRAR is a regional student level conference, focusing on introducing and educating local students into research, designed to accept research work in progress and maximize constructive feedback. It is expected that many submissions describe partial results obtained during semester student projects or incremental thesis work progress. This limitation was even more visible in the 2020 edition as team student projects were heavily limited due to the extended Covid-19 lockdown.

Since, based on the FCRAR mission, all relevant submissions from Florida students are commonly accepted, the reviews are not focused on acceptance/rejection decisions, but on providing constructive feedback. To improve the quality of the reviews, this year reviewers were incentivized to contribute more substantial feedback by giving them credit through on open peer-review process.

By open peer-review in FCRAR 2020 we mean that the reviews for submissions are made available for everybody, being published in these proceedings together with the draft article for which they were written. Reviews are published together with the names of the reviewers such that reviewers can get credit for the ideas and contributions that they provide, guaranteeing their intellectual property for consistent feedback.

The original submissions, relevant as basis for these reviews, and as basis for intellectual property claims in case of conflict with reviewers, are published in the Appendix of the proceedings. This provides material evidence for the case where a reviewer claims his feedback introduced a new patentable idea that the authors believe was already suggested in the original submission. It also helps assess the quality of the review by independent 3rd parties, thereby encouraging quality reviewing.

We also publish the answer to reviews (review rebuttals), as provided by the authors of the reviewed article, to help correct misconceptions potentially introduced by the review, and further encourage and acknowledge quality reviews.

The reviews are themselves reviewed by the corresponding rebuttal of the authors, and can therefore be considered as peer-reviewed contributions. Reviewers were allowed for one week to withdraw individual paragraphs in a review, after seeing the corresponding rebuttal, providing a way to clean unimportant or erroneous suggestions. Authors were subsequently also allowed for one week to withdraw paragraphs in their rebuttals, or even the entire submission, after seeing the cleaned final versions of the reviews.

The scoring mechanism used in the review procedures of this conference differs significantly from current practice. We decide to link the provided score with objective measures, in particular to utility in dollars, as a suggestion to the authors of how much funding they could ask for finalizing their research. Each reviewer was able to provide two unbounded numerical scores. The first score stands for the Utility for the FCRAR Community and the second score stands for Utility for the Society as a Whole. The instructions requested that these numbers should be expressed in terms of dollars, and should represent the amount of funding the reviewer considers as worthy for the given contribution (if the work would not have been done yet). It was possible to input a negative value if the work was deemed harmful. The utility value zero was not accepted.

At the previous event when this approach was used, DC2013, the instructions requested that these numbers should be expressed in terms of *thousands of dollars*, and that approach should probably be adopted in future editions as it resulted in a smaller number of confused reviewer reports.

Utilities assigned to submissions ranged from \$1 to \$10,000,000 with median \$12.5, and mode \$1 (9 reviews). **The Highest Estimated Utility Paper Award** goes to *"3D Localization of Defects in Facility Inspection"* by Nicholas Califano and David Murakami, which proverbially was estimated at both extremes evaluations, being evaluated at \$10,000,000 by one reviewer and at \$1 by its other reviewer. The authors replied in kind by estimating the utility \$10,000,000 to the first review, **The Highest Estimated Utility Review Award** going to Roussi Roussev. Offering these two awards in the future should be discouraged as it would lead to inflation warping the incentives for fair reviewing. The median and mode for reviews utility estimates was \$50.

In general one can observe a *retaliatory phenomena* in evaluations of utilities of reviews, where most reviews were evaluated at the same value as the utility they had assigned to the paper, with the exception of "Patient Walker Application and Human Subject Tests with an Assistive Robot" (\$100000 to \$50), "A Review of Rapidly Exploring Random Tree Path Planning Algorithms" (\$1 to \$1000), and "Trained Probabilistic Models for the NAO Robot in a Labyrinth" (\$1 to \$200).

The more significant difference to most other conferences is that the whole review and rebuttal exchange is made available. Unlike with journals, the short time frame of the review for a conference limits the amount of rounds of review/rebuttal exchanges that can be documented. As such, a single round is allowed, and reviewers cannot change their original review except for withdrawing whole paragraphs.

One can make a distinction between reviewing and paper acceptance, by seeing reviewing of research articles in the same way as reviewing of items on Amazon and eBay. There, a review describes good and weak points of an item. It is up the reader to interpret the review and to decide whether to buy the item. Applying this to conference reviewing, the reviews and their rebuttals have to be provided, ranking the articles by their perceived utility, and it is up to the reader to decide whether she wants to proceed with reading or not.

The concept of paper rejection is typically needed with limited resources (limited time available in conferences, and limited space in paper journals), but it is less justified when proceedings are electronic or when there exists space for oral presentation or posters. Rejection just for the sake of attaining an arbitrary rejection threshold is otherwise in conflict with freedom of speech principles, and encourages reckless reviewing (as described in the game theoretic studies [IEEE-IIB vol 13. 2012]). Nevertheless, an author can withdraw her articles rather than have them published with reviews that point to significant errors.

A somewhat similar peer-review mechanism was used in the Decentralized Coordination Workshop (DC2013), with the difference that there reviewers were allowed to optionally review blindly, i.e. without seeing the names of the paper authors, if they wanted to improve their impartiality. There the whole process was manual while here the process was fully automated using the `fcrar2020.fit.edu/submissions.php` system implemented at Florida Tech. While open peer-review is being used in other research communities, such as Biology and Design, those mechanisms differ from the ones employed here.

Program committee members were allowed to bid on reviewing any number of articles that they were interested in reviewing. Some committee members bade on as many as 5 articles while others were eventually not available for any review. Some reviewers bad on many articles but eventually reviewed none.

During the bidding period, reviewers had access only to titles and abstracts of the submissions. The submission with the highest number of bids is offered **The Most Appealing Paper Abstract Award**. This article is: "Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation" by Alexander Frye and Hadi Fekrmandi. Nine submissions were withdrawn, and only one of the submitted articles was presented without having attracted reviews.

A social network describing the shared interest of reviewers as revealed by their reviewing and bids, is depicted on the proceedings cover, based on the algorithm introduced by Song Qin at DC2013.

Marius Silaghi, FCRAR2020 Chair

Bids Cast for Reviewing Submissions

Here we list the reviewing bids in the order of their submission.

- A Review of Rapidly Exploring Random Tree Path Planning Algorithms
  Reviewers: **Roussev Roussi, Ribeiro Eraldo**, Bidders: Filipovich Roman, Savaux Julien
- Trained Probabilistic Models for the NAO Robot in a Labyrinth
  Reviewer: **Barták Roman**, Bidder: Alhamed Khalid
- Machine Learning for Development of Fault Identification and Risk Management (FIRM) of Autonomous Underwater Vehicles (AUVs)
  Bidders: Denzinger Joerg, Sevil Hakki Erhan, Eskridge Thomas
- Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation
  Reviewers: **Sevil Hakki Erhan, Filipovich Roman, Wilde Markus**, Barták Roman, Eskridge Thomas
- Compliant Joint Actuated Via Cables
  Reviewer: **Haghshenas-Jaryani Mahdi**, Bidder: Fekrmandi Hadi, Wilde Markus
- Anomaly Detection in the Joint Angle Sensors of a Snake Robot
  Reviewer: **Morris Melissa**, Bidder: Barták Roman
- Vision-based decision system for humanoid robot
  Reviewer: **Barták Roman**, Bidder: Eskridge Thomas
- 3D Localization of Defects in Facility Inspection
  Reviewers: **Ribeiro Eraldo, Roussev Roussi**, Bidders: Filipovich Roman
- Patient Walker Application and Human Subject Tests with an Assistive Robot
  Reviewer: **Silaghi-Horisberger Anne Melody**, Bidders: Morris Melissa, Eskridge Thomas
- Wave propagation in 3D printed parts with different internal characteristics
  Bidder: Fekrmandi Hadi
- Cold Spray Additive Manufacturing - Process Review
  Reviewer: **Morris Melissa**
- A Neurofeedback-Driven Humanoid to support Deep Work
  Reviewer: **Filipovich Roman**
- Revisiting the Pull Test for the Measurement of Static Coefficient of Friction
  Reviewers: **Sevil Hakki Erhan, Morris Melissa, Wilde Markus**

Bids have also been cast for the next submissions that were eventually neither presented not withdrawn: *A Multithreaded Paradigm for Swarm Computing and Redundancy, Plan for an intelligent robotic and wearable based hospital sanitization support system*, and *Concept Development for Intelligent Transient Communities.*

Contents

Conference Committees

Conference Chairs

- Silaghi, Marius [msilaghi@fit.edu]; Florida Tech

Senior Committee

- Stansifer, Ryan [ryan@fit.edu]; Florida Tech
- Wilde, Markus [mwilde@fit.edu]; Florida Institute of Technology
- Eskridge, Thomas [teskridge@fit.edu]; Florida Institute of Technology
- Gutierrez, Hector M [hgutier@fit.edu]; Florida Institute of Technology

Track: Main

- Sevil, Hakki Erhan [hsevil@uwf.edu]; University of West Florida
- Eskridge, Thomas [teskridge@fit.edu]; Florida Institute of Technology
- Filipovich, Roman [romanmail@apple.com]; Apple
- Morris, Melissa [Melissa.Morris1@erau.edu]; Florida Polytechnic University
- Denzinger, Joerg [denzinge@cpsc.ucalgary.ca]; University of Calgary
- Silaghi, Marius [msilaghi@fit.edu]; Florida Tech
- Ni, Zhen [zhenni@fau.edu]; FAU
- Gutierrez, Hector M [hgutier@fit.edu]; Florida Institute of Technology
- Fekrmandi, Hadi [hadi.fekrmandi@sdsmt.edu]; South Dakota School of Mines and Technology
- Song, Sang-Eun [s.song@ucf.edu]; University of Central Florida
- Tansel, Ibrahim [tanseli@fiu.edu]; Florida International University
- Masory, Oren [masoryo@fau.edu]; FAU
- Stansifer, Ryan [ryan@fit.edu]; Florida Tech
- Wilde, Markus [mwilde@fit.edu]; Florida Institute of Technology
- Morris, Melissa [morrim44@erau.edu]; Embry-Riddle Aeronautical University

Track: Hardware

- Morris, Melissa [Melissa.Morris1@erau.edu]; Florida Polytechnic University
- Gutierrez, Hector M [hgutier@fit.edu]; Florida Institute of Technology
- Integlia, Ryan [integlia@ieee.org]; The Citadel
- Wilde, Markus [mwilde@fit.edu]; Florida Institute of Technology
- Morris, Melissa [morrim44@erau.edu]; Embry-Riddle Aeronautical University

Track: Intelligence

- Eskridge, Thomas [teskridge@fit.edu]; Florida Institute of Technology
- Savaux, Julien [savaux@gmail.com]; University of Valenciennes
- Alhamed, Khalid [hamedk@ipa.edu.sa]; Institute of public administration
- Alqahtani, Abdulrahman [msalthaqfan@nu.edu.sa]; Najran University

Program Committee

- Stansifer, Ryan [ryan@fit.edu]; Florida Tech
- Wilde, Markus [mwilde@fit.edu]; Florida Institute of Technology
- Elmane, Shakre [selmane2015@my.fit.edu]; FIT
- Eskridge, Thomas [teskridge@fit.edu]; Florida Institute of Technology
- Gutierrez, Hector M [hgutier@fit.edu]; Florida Institute of Technology
- Denzinger, Joerg [denzinge@cpsc.ucalgary.ca]; University of Calgary
- Ni, Zhen [zhenni@fau.edu]; FAU
- Fekrmandi, Hadi [hadi.fekrmandi@sdsmt.edu]; South Dakota School of Mines and Technology
- Song, Sang-Eun [s.song@ucf.edu]; University of Central Florida
- Morris, Melissa [morrim44@erau.edu]; Embry-Riddle Aeronautical University
- Masory, Oren [masoryo@fau.edu]; FAU

Track: Hardware

- Sevil, Hakki Erhan [hsevil@uwf.edu]; University of West Florida
- Filipovich, Roman [romanmail@apple.com]; Apple
- Alhamed, Khalid [hamedk@ipa.edu.sa]; Institute of public administration
- Morris, Melissa [Melissa.Morris1@erau.edu]; Florida Polytechnic University

Track: Intelligence

- Barbu, Adrian [abarbu@stat.fsu.edu]; Florida State University
- Hasanain, Ahmad [ahasanain2014@my.fit.edu]; FIT
- Ribeiro, Eraldo [eribeiro@fit.edu]; Florida Institute of Technology
- Weekes, Troy [tweekes1998@my.fit.edu]; Florida Tech
- Elmane, Shakre [selmane2015@my.fit.edu]; FIT
- Roussev, Roussi [rroussev@splunk.com]; Splunk
- Savaux, Julien [savaux@gmail.com]; University of Valenciennes
- Filipovich, Roman [romanmail@apple.com]; Apple
- Alhamed, Khalid [hamedk@ipa.edu.sa]; Institute of public administration

Track: Main

- Elmane, Shakre [selmane2015@my.fit.edu]; FIT
- Doe, John [john.doe@silaghi.org]; Florida Schools
- Haghshenas-Jaryani, Mahdi [mahdihj@nmsu.edu]; New Mexico State University
- Barták, Roman [bartak@ktiml.mff.cuni.cz]; Charles University
- Denzinger, Joerg [denzinge@cpsc.ucalgary.ca]; University of Calgary
- Rawi, Musab [alrawi83@yahoo.com]; Management and Science University (Malaysia)
- Kepuska, Veton [vkepuska@fit.edu]; Florida Institute of Technology
- Weekes, Troy [tweekes1998@my.fit.edu]; Florida Tech
- Integlia, Ryan [integlia@ieee.org]; The Citadel
- Sevil, Hakki Erhan [hsevil@uwf.edu]; University of West Florida
- Roussev, Roussi [rroussev@splunk.com]; Splunk
- Savaux, Julien [savaux@gmail.com]; University of Valenciennes
- Barbu, Adrian [abarbu@stat.fsu.edu]; Florida State University
- Filipovich, Roman [romanmail@apple.com]; Apple
- Alhamed, Khalid [hamedk@ipa.edu.sa]; Institute of public administration
- Morris, Melissa [Melissa.Morris1@erau.edu]; Florida Polytechnic University

Part I

Track Intelligence

3D Localization of Defects in Facility Inspection

Nicholas Califano
University of Central Florida

David Murakami
NASA Ames Research Center

ABSTRACT

Wind tunnels are crucial facilities that support the aerospace industry. However, these facilities are large, complex, and pose unique maintenance and inspection requirements. Manual inspections to identify defects such as cracks, missing fasteners, leaks, and foreign objects are important but labor and schedule intensive. Our goal is to utilize small Unmanned Aircraft Systems (sUAS) and computer vision-based analysis to automate the inspection of the interior and exterior of NASA's critical wind tunnel facilities. We detect missing fasteners as our defect class, and detect existing fasteners to provide potential future missing fastener sites for preventative maintenance. These detections are done on both 2D raw images and in 3D space to provide a visual reference and real world location to facilitate repairs. A dataset was created consisting of images taken along a grid-like pattern of an interior tunnel section in the AEDC National Full-Scale Aerodynamics Complex (NFAC) at NASA Ames Research Center. Our method uses object detection to create image level bounding boxes of the fasteners and missing fasteners, then uses photogrammetry to create a mapping from 2D image locations to 3D real world locations. The image level bounding boxes and the 2D to 3D mapping are then combined to determine the 3D location of the defects. We describe the data collection, photogrammetry, and computer vision techniques used for object detection as well as a quantitative analysis of the method.
Keywords: Small Unmanned Aircraft Systems (sUAS), Object Detection, 3D Localization.

1. Introduction

The NFAC facility is the largest wind tunnel in the world, and supports both military and commercial programs in the aerospace industry. Maintenance of the facility requires routine inspections to be performed by technicians, but it is nearly impossible to perform frequent, thorough, manual inspections of the NFAC facility as there is more than one million square feet of interior surface area. Additionally, the NFAC facility surfaces contain complex geometry and ceilings that vary in height from 50' to 250', making inspections of some areas particularly difficult.

Our goal is to use drones to capture images along grid-like patterns of the facility surfaces, and to automate the process of defect detection with computer vision-based analysis. The system must be able to provide the 3D location of the defect, as well as images containing the defect for visual reference. The proposed method in this paper outlines a pipeline to detect defects in 3D space from the set of images taken in the facility.

Figure 1: NFAC Facility at NASA Ames Research Center
Source: [1]

2. Related Works

Photogrammetry. Photogrammetry is a well-studied and accurate method to extract 3D spatial dimensions from a set of 2D images taken at various locations [2]. Through the use of classical computer vision methods, object recognition can be used to match local features between images. Given the camera poses, camera parameters, and optical parameters, the matched features can be used to determine the 3D location of the object [3]. Photogrammetry creates keypoints, each of which map a 3D point to the 2D pixel locations of the images in which that 3D point is visible. However, this method alone is not able to determine the class of the objects recognized and therefore cannot determine if a recognized object is a defect.

Object Detection. There are two main fields of computer vision that determine the pixel location of an object instance in an image: object detection and instance segmentation. Object detection creates a bounding box around each object instance while instance segmentation creates a pixelwise mask of each object instance. Since the application of wind tunnel facility inspections is relatively niche, a custom dataset was required. Manually labeling datasets is expensive in terms of monetary cost, time, and effort [4, 5]. Bounding box labels can be created using significantly less man hours than pixelwise masks. Additionally, bounding box level detections are more than sufficient for visual reference and to perform 3D localization. As such, object detection was selected as the preferred method.

All object detection methods are composed of three main components: a region proposer, a feature extractor, and a classifier. The

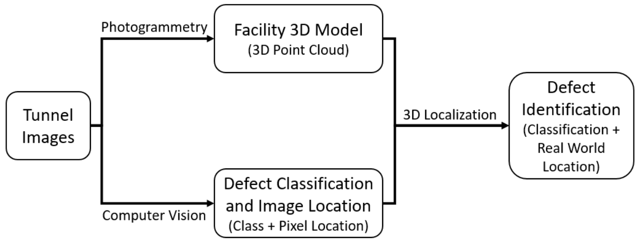


Figure 2: Method Pipeline

region proposer produces potential object locations in an image in the form of regions of interest. This method can ignore contextual information and procedurally generate bounding boxes based on the image size [6, 7], or use contextual information to generate bounding boxes [8, 9, 10, 11, 12]. The feature extractor creates high density information related to the features of the image. Classical machine learning methods are based on hand-crafted features such as gradients, color, etc [13, 14, 15, 16]. Deep learning object detectors typically make use of convolutional neural networks for feature extraction [11, 6, 7]. The classifier determines the class of the object based on the extracted features. Commonly used classical methods for classification include Bayes classification [17], k-nearest-neighbors [18], support vector machines [19], and decision trees [20]. Fully-connected neural networks have gained popularity recently [21, 22] and most state-of-the-art methods [23, 11] employ them for classification. Object detectors can be further divided between region proposal detectors, and regression detectors [21, 22].

Region proposal detectors have some method that creates regions of interest to be classified. This method can use classical computer vision [9] such as graph based image segmentation [8, 24] or a deep learning method such as a Region Proposal Network (RPN) [11, 12]. Feature extraction and classification is then performed for each region to determine the class of the object captured by the region. Regression methods overlay a regular grid on the image rather than having a distinct region proposal step. Feature extraction is performed on the entire image, and each grid space is classified [6, 7]. For both types of classifiers, bounding box regression is performed to improve localization performance, and Non-Maximum Suppression (NMS) is performed to remove redundant classifications [9, 10, 11, 6, 7].

3D Object Detection. Autonomous driving is a common application of 3D object detection [25, 26, 27], however this application is distinct from ours. Often times these systems rely on a LiDAR for point cloud generation. Due to the weight constraints of an aerial vehicle, our system does not implement a LiDAR. Additionally, the image capture locations in these systems are positioned along a parametric line along which the data was collected. These systems are able to process the input images similarly to a video stream, and propagate information between sequential frames. In our application, image capture locations are positioned along a parametric surface that follows the inspection surface rather than a one dimensional parametric line.

The most similar method to ours available is [28], but their method is intended for traffic sign detection. Their method was again intended for frames captured along a parametric line, but could be applied to a parametric surface with slight modifications. The main improvement of our method over [28], is the implementation of more powerful algorithms for point cloud generation and object detection.

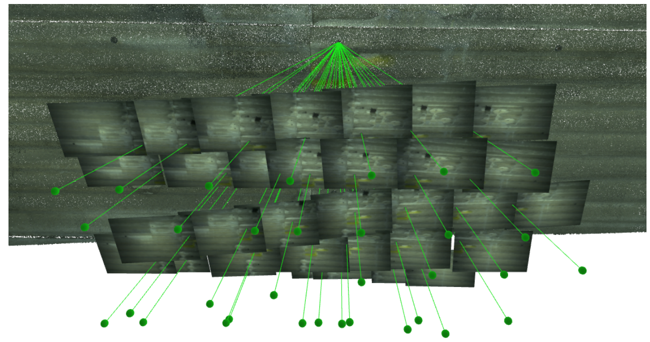


Figure 3: Tracing Pixel Locations of an Object to the 3D Location

3. Method

Our method consists of three stages as shown in Figure 2. The photogrammetry stage uses the set of input images to generate a 3D point cloud. The point cloud consists of keypoints which map a 3D point to the 2D pixel points in the images in which the 3D point is visible. The computer vision stage uses the set of input images and generates object detections in the form of image level bounding boxes with an associated class. The 3D localization stage uses the object detections and the keypoints to localize the objects in 3D space.

3.1 Photogrammetry

Pix4D [29] was used for the photogrammetry purposes of our method. Pix4D is able to process a set of images with no prior knowledge of the camera positions. This is highly desirable as the recorded position of the camera when the image is captured is subject to noise and drift, or may be completely unavailable in the worst cases. Pix4D generated an average of almost 3 thousand 3D keypoints per image for our dataset, and each of the keypoints was visible in an average of 4 images.

3.2 Computer Vision

Object detection was the selected computer vision task to localize the pixel locations of the defects in the images. Our method uses a deep learning method for object detection due to the significant increase in performance of deep learning methods observed in recent years [21, 22]. The current state-of-the-art methods use convolutional neural networks, and can be divided between two main groups, the region proposal methods and regression methods. Region proposal methods are typically slower than regression methods due to the evaluation of each region individually. However, region proposal methods typically have better performance than regression methods, especially on small objects [30]. Due to the nature of facility inspections, almost all of the objects are considered to be small to medium sized since they are smaller than 128 x 128 pixels. A plot of the pixel size of all object instances is shown in Figure 4. As such, the implemented architecture for our method was based on the R-CNN family of detectors [9, 10, 11, 12].

Our computer vision method uses the object detection portion of a ResNet18 Mask R-CNN [12] with slight modifications to the priors and training scheme. We did not implement the mask branch of Mask R-CNN, and uses only the class and bounding box outputs. Due to the differences in dataset, we made modifications to the anchor box priors, type of data augmentation, training parameters, and implemented class balancing techniques.

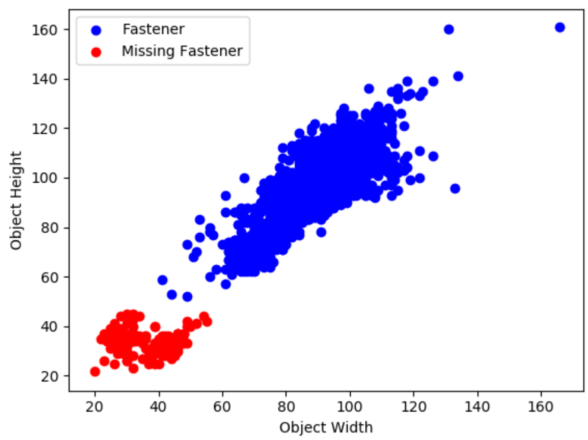


Figure 4: Image Object Instance Pixel Size

The anchor box sizes and aspect ratios used in the original Mask R-CNN implementation, and most object detectors, are rather large for our dataset. Our anchor box priors were selected using k-means clustering similarly to YOLOv3 [7]. We initialized two clusters per class, and maximized the average Intersection over Union (IoU) across all samples using k-means clustering.

During training the only form of data augmentation used by Mask R-CNN was horizontal flipping. This is likely because other forms of data augmentation would produce unrealistic images. Our dataset is composed of facility surfaces, which can be augmented more aggressively and still produce realistic images. As such, we implemented random horizontal flipping with probability of 50%, and random rotation in 90 degree increments, each with 25% likelihood.

Mask R-CNN used 16 images per batch during training, but in our experiments we used 1 image per batch as done in Faster R-CNN [11]. Additionally, Mask R-CNN uses 0.7 as the IoU threshold to classify an anchor box as a positive sample and an RPN batch size of 256 samples per image. For the classifier, Mask R-CNN assumes a positive to negative ratio between 1:3 and 1:1. Unlike the datasets used in object detection challenges [31, 32], our dataset is not object-centric and so with those parameters our positive to negative ratio was much lower. To increase the number of positive samples for the RPN, we set the objectness IoU threshold to 0.5. Even so, our dataset produces a low concentration of positive samples so as to better preserve the positive to negative ratio, we reduced the RPN batch sizes to 128 per image. For the classifier, we allow the positive to negative ratio to vary from 1:7 up to 1:1 by varying the classifier per image batch size and capping it at a maximum of 64 samples per image. However, in practice the ratio rarely gets above 1:3 for the classifier. Finally, since our dataset is significantly smaller than the that of object detection challenges, our training schedule was shorter than that of Mask R-CNN. We trained with a learning rate of 0.001 for 30k iterations, as this was the point that performance degraded on the validation data.

Our dataset featured severe class imbalance between the fasteners and missing fasteners. The number of missing fastener samples was about one eighteenth that of the number of fastener samples, and so we oversampled the images with missing fasteners eighteen times so that the same number of training iterations performed on fasteners and missing fasteners was equal. Additionally, since the fasteners were much larger than the missing fasteners, the fasteners had roughly five times the number of anchor boxes above the objectness IoU threshold per instance than missing fasteners. To

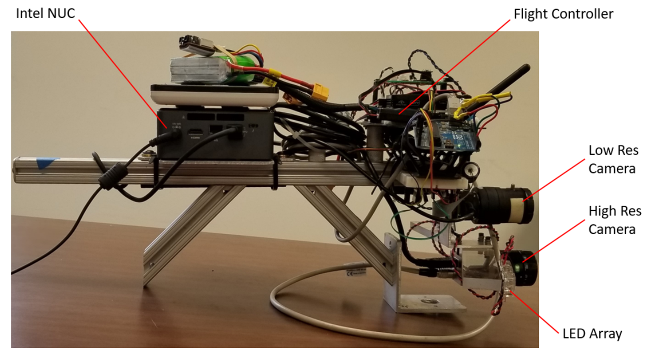


Figure 5: Handheld Data Collection System

combat this, during training we weighed the missing fasteners to be roughly five times as important as fasteners. Both of these efforts improved the performance of the model on missing fasteners.

3.3 3D Localization

Once the point cloud and pixel level object locations are created, the objects can be localized in 3D space. At first, the 2D pixel location of each keypoint is considered. A keypoint is associated with an object detection if it lies within the detection's bounding box. Each keypoint is associated with several images, and when two or more object detections share the same keypoint, the object detections are grouped together into an object set. Once all keypoints have been associated with object detections, and the all object detections grouped into object sets, the 3D location of the keypoints is considered. The keypoints associated with an object set are then clustered with mean shift clustering to determine an average 3D location of the keypoints. Mean shift was selected as it finds the maxima of the spatial density. A maxima in spatial density indicates that many keypoints are 'confident' that the object is located in that position. Figure 3 shows an example of the trace lines from the camera position, through the image plane at the position of the fastener, to the 3D location of a fastener.

4. Experiment

4.1 Dataset

The dataset collected was from a section of the interior NFAC walls over a 20 meters by 2 meters area using the handheld system shown in Figure 5. The images were collected at a distance of 1 meter from the wall, and captures on a grid-like pattern along the wall. The grid was spaced such that each object appears in 6-7 images per row, and in 4-5 rows of images. Objects along the border of the inspection area will appear in fewer images per row and/or fewer rows of images. The image capture locations were supervised by a Simultaneous Localization And Mapping (SLAM) algorithm to ensure that the spacing between capture locations, and the distance between the cameras to the wall was within tolerance.

Two types of labels were created for the dataset. First, the image object labels were created in the PASCAL VOC [31] format. Fasteners and missing fasteners instances were given a bounding box and class label. An example of objects from our dataset is shown in Figure 6. The second type of label created was the object set label. This label was the set of all image bounding boxes associated with the real world object.

Objects in the training, validation, and testing partitions must be distinct. Each real world object cannot appear on more than

Figure 6: Object Examples with Highlighted Fastener (Blue) and Missing Fastener (Red)

one of these partitions, as it would cause data leakage and unfairly increase the performance of the model. To ensure that each object appears in no more than one partition, the object sets were divided between the training, validation, and testing partitions. Objects along the border of the inspection area were not places in the validation or testing set as the reduced number of image instances is not representative of a full scale facility inspection. The object sets were divided in an 64/16/20 split between the training, validation, and test partitions respectfully. Images containing multiple objects were cropped into subimages such that each subimage contains objects of only one partition.

The training set contained 110 fastener object sets and 4 missing fastener object sets. Many of the training object sets were on the border of the inspection and so they did not contain the full 24 to 35 images per object set. The validation set contained 15 fastener object sets and 1 missing fastener object set. The test set contained 21 fastener object sets and 1 missing fastener object set. The object sets in both the validation and test set contained the full 24 to 35 images per object set.

4.2 Results

No analysis of the accuracy of the photogrammetry was performed as Pix4D provides its own analysis. We use frame-mAP and video-mAP as defined by [33] to quantitatively evaluate our results. Frame-mAP is used to evaluate the object detector and measures the mean Average Precision (mAP) on the image level labels. For video-mAP, we use the ground truth object set labels in place of actual video labels. Video-mAP measures the mAP of the full method per object set. For both metrics we use an IoU threshold of 0.5 to be considered a true positive. The results are shown in Table 1.

On the per frame level, our method had an average precision of 92.1% when detecting fasteners, and 31.0% when detecting the missing fasteners in the test set. On the per object level, the object detector had an average precision of 86.4% when detecting fastener object sets and 0.00% when detecting missing fastener object sets. For the single missing fastener object set in the test set, 37% of the image instances were properly detected. Unfortunately since this is less than 50% of the image instances, it was not considered an object set detection for the video-AP. The decrease in performance on the missing fasteners is likely due to the both their smaller size and the smaller number of samples to train upon.

Unfortunately, due to the relatively small amount of object sets, the results are not conclusive. The results show potential as we

Class	Frame-AP	Video-AP
Fastener	92.1	86.4
Missing Fastener	31.0	0.00
Mean	61.5	43.2

Table 1: mAP Performance

were able to locate most of the fastener objects on both the image level and in 3D space. However, in the case of missing fasteners there was only one object set to test upon, and so the results are unreliable.

5. Conclusion

Our method performed well and our results are promising. This method has the potential to enable quick and inexpensive inspection of large wind tunnel facilities. We were able to successful perform object detection in the 2D raw images and localize the defects in 3D space. Additionally, the inference time was on the order of seconds per image using a single Tesla V100 GPU. With a reasonable number of GPUs the entire facility can be processed over a week. More importantly, this process of utilizing drones requires significantly less tunnel downtime than with manual inspections, and will have easier access to the difficult to reach areas of the tunnel. We intend to continue this project and collect significantly more data to improve the performance of the model, and to properly test and evaluate this method.

6. Acknowledgements

We are grateful for the support from Fluid Mechanics Lab at Ames Research Center for sponsoring this project, AEDC for allowing us access to the NFAC facility, and the NASA Advanced Supercomputing (NAS) Division at Ames Research Center for supporting us with the required computational resources.

References

[1] J. Dino, “NASA - National Full-Scale Aerodynamics Complex (NFAC).”

[2] David W. James, Juergen Eckermann, Fawsi Belblidia, and Johann Sienz, “Point cloud data from Photogrammetry techniques to generate 3D Geometry,” *Proceedings of the 23rd UK Conference of the Association for Computational Mechanics in Engineering 8*, no. April, p. 5, 2015.

[3] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[4] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, “LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop,” 2015.

[5] S. Vijayanarasimhan and K. Grauman, “What’s it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations,” *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, vol. 2009 IEEE, no. June, pp. 2262–2269, 2009.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, “Ssd: Single shot multibox detector,” in *European conference on computer vision*, pp. 21–37, 2016.

[7] J. Redmon and A. Farhadi, “YOLOv3: An Incremental Improvement,” 2018.

[8] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[10] R. Girshick, "Fast R-CNN," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 1440–1448, 2015.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *arXiv preprint arXiv:1703.06870*, pp. 2961–2969, 2017.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*, pp. 404–417, 2006.

[15] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: An efficient alternative to SIFT or SURF.," in *ICCV*, vol. 11, p. 2, 2011.

[16] M. Brown, R. Szeliski, and S. Winder, "Multi-image matching using multi-scale oriented patches," in *CVPR*, pp. 510–517, 2005.

[17] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[18] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 4, pp. 325–327, 1976.

[19] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[20] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Routledge, 2017.

[21] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object Detection With Deep Learning: A Review," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2019.

[22] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and Others, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311, 2017.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[24] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[25] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3D Object Detection for Autonomous Driving," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2147–2156, 2016.

[26] C. Xiaozhi, M. Huimin, W. Ji, L. Bo, and X. Tian, "Multi-View 3D Object Detection Network for Autonomous Driving," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1915, 2017.

[27] Y. Zhou and O. Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, 2018.

[28] V. Balali and M. Golparvar-Fard, "Recognition and 3D Localization of Traffic Signs via Image-Based Point Cloud Models," *Congress on Computing in Civil Engineering, Proceedings*, pp. 206–214, 2015.

[29] "Professional photogrammetry and drone mapping software | Pix4D,"

[30] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[31] M. Everingham, L. Van~Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results." http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[33] G. Gkioxari and J. Malik, "Finding action tubes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 759–768, 2015.

Review of Paper #105

Roussev, Roussi
Splunk
rroussev@splunk.com

Estimated Utility of:
"3D Localization of Defects in Facility Inspection"
Utility to Community: $10000000
Utility to Society: $10000000

Living a few miles from Ames, working daily with an engineer who has spent a significant part of his career in those wind tunnels and repeatedly hearing how computers have mostly replaced the need for expensive equipment, I found this paper very interesting. It proposes methods for automated inspection of their tunnels. It is well presented and the solution looks promising.

At the same time, it will be nice to add more details on the problem to truly appreciate the solution. How often is manual inspection done and how long it takes? The classes of issues found, their frequency and criticality level. Are the Ames fasteners special or one can transfer from/to other (generative) models and datasets? Why is an array of stationary cameras not a good idea (there must be a good reason in this context)? Is the additional complexity of 3d localization really important to get right from the beginning? Would a multi-pass inspection be appropriate?

Further Details of Inspection Problem

Califano, Nicholas
University of Central Florida
`nickcalifano@knights.ucf.edu`

Answer To Review:
"Review of Paper #105"
Utility Review: $10000000

Great questions, and I'll address them individually, but a bit out of order:

Why is an array of stationary cameras not a good idea (there must be a good reason in this context)?

Being a large wind tunnel, the NFAC facility poses some unique challenges and is not the same as an equally sized warehouse. The facility features complex geometry and some hard to reach places, so setting up stationary or even sweeping cameras to cover all areas would be quiet difficult. Second, the facility has over one million square feet of interior surface area, and so it would require many cameras.

Additionally, unlike a typical warehouse the 'empty' space serves a purpose. Airflow through the empty spaces must be unobstructed to remain laminar. Turbulent flow would result in poor quality experimental results and setting up cameras everywhere, even in streamline bubbles, would be problematic.

Drones solve this by being small and agile enough to reach all areas, and do not permanently occupy space in the tunnel.

How often is manual inspection done and how long [does] it takes? The classes of issues found, their frequency and criticality level.

The NFAC facility does not currently have regular, thorough, inspections performed. An estimate of 100 work days per year of man hours would be required to inspect the facility. During that time the tunnel would need to be inoperable, and no testing could be performed. Additionally, it would require cutting holes in the ceiling and attaching ropes increasing maintenance costs past that of just the man hours.

The current inspections are done with engineers patrolling the facility with flashlights and spotter scopes. Minor issues are not infrequent, and are logged to be revisited so that they can be surveilled if they become major.

The main classes of issues are missing fasteners, cracks/spalling, leaks, and foreign object debris. All of these issues can range from minor to major severity. The biggest issue being that since it is a wind tunnel, if anything becomes loose it will travel with the airflow potentially damaging the drive system, experiment, or other tunnel structures. Airflow in the smaller test section can be upwards of 300 mph, so even a small bolt can be a destructive projectile. Cracks/spalling and missing fasteners can cause panels to come loose. Leaks can cause corrosion which also allows for components to come loose. Foreign objects debris can travel in the tunnel by themselves.

Are the Ames fasteners special or one can transfer from/to other (generative) models and datasets?

The Ames fasteners are not special, but the background of the tunnel is relatively unique. Likely in the future if this method is used for other facilities, the model would be pretrained on the NFAC dataset and finetuned on the new facility or vice versa.

Is the additional complexity of 3d localization really important to get right from the beginning?

Identifying defects solely in 2D images is a bit trivial given the current state of computer vision. While it may present a new application for these object detection methods, nothing new as far as the method would need to be developed.

The goal of our method is to not only find defects, but localize them in 3D. That way we could direct maintenance personal to the location of the defect, and supply images so that they can prepare the equipment to address the issue. This would allow for the minimum tunnel downtime.

Would a multi-pass inspection be appropriate?
Absolutely. A more long-term goal of the project is to perform these inspections more regularly. Processing of the images will be done offline, and so the only required downtime is for the images to be taken. Each scan of the facility would give another chance for missed defects to be caught.

Review of 3D Localization of Defects in Facility Inspection

Ribeiro, Eraldo
Florida Institute of Technology
eribeiro@fit.edu

Estimated Utility of:
"3D Localization of Defects in Facility Inspection"
Utility to Community: $1
Utility to Society: $1

The paper presents an application of using drones, AI, and computer vision to the problem of inspecting large industrial environments. The work reports a method to inspect a wind tunnel. Inspection focuses on detecting small defects and missing parts on roofs and walls (apparently).

It is a well-written paper with a clear application. While the work is a preliminary implementation, experimental results show that the method works and is promising.

A Review of Rapidly Exploring Random Tree Path Planning Algorithms

Jacqueline Jermyn, Rodney Roberts

FAMU-FSU College of Engineering
2525 Pottsdamer Street
Tallahassee, Florida, 32310

jmj15c@my.fsu.edu, rroberts@eng.famu.fsu.edu

ABSTRACT
Path planning algorithms are robot navigation techniques that determine a robot's route from its beginning to its goal configurations. The Rapidly Exploring Random Tree (RRT) algorithm is a sampling-based path planning algorithm, which begins at an initial configuration and builds a tree containing vertices and edges until it reaches its goal configuration. It is extensively implemented because of its simplicity, but it does not determine optimal paths. As a result, various modifications have been made to improve its performance. This paper reviews simulations to evaluate the performance of the RRT, Rapidly Exploring Random Tree Star (RRT*), and Informed RRT* algorithms.

Keywords
Informed Rapidly Exploring Random Tree Star (Informed RRT*), Path Planning, Rapidly Exploring Random Tree (RRT), Rapidly Exploring Random Tree Star (RRT*)

1. INTRODUCTION
In a robotic navigation scenario, a robot's environment is also known as the configuration space C to include free space C_{free} and obstacle space C_{obs} [1]. Free space encompasses all spaces that the robot can occupy. Obstacle space contains obstructions that restrict a robot's movements. Consequently, free space and obstacle space are complements of each other. Path planning algorithms determine routes to move a robot from a start configuration $\mathbf{q}_0$ to a goal configuration $\mathbf{q}_G$ in free space C_{free} without colliding with the obstacle space C_{obs} [19]. Figure 1 depicts a generalized path planning environment showing free space, obstacle space, the initial configuration, and the goal configuration.

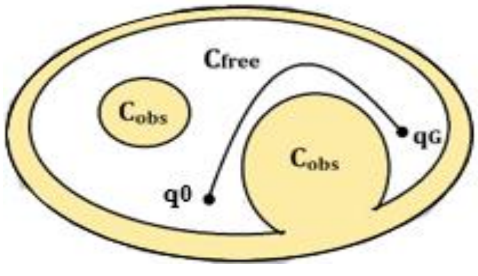


Figure 1. Generalized Path Planning Environment

Path-planning techniques are divided into combinatorial and sampling-based planning methods, as shown in Figure 2 [1], [29]. Combinatorial methods build structures in the configuration space that discretely and completely obtain all the necessary data to plan a path [19]. These algorithms are complete because they will either find a path, or they will decide that a path does not exist [18]. These methods also determine an exact representation of the free space and build a roadmap, which is a graph that characterizes free space [30]. A path on the roadmap can then be generated by implementing a graph search algorithm, such as breadth-first search, depth-first search, or Dijkstra's algorithm [3], [30].

Types of combinatorial path planning algorithms include visibility graphs, cell decomposition methods, and Voronoi diagrams [1]. Visibility graphs are networks of intervisible locations consisting of vertices and edges. Vertices include corners of obstacles, the initial configuration, and the goal configuration, while edges consists of collision-free connections between vertices of obstacles.

Cell decomposition techniques discretize the environment into cells to include exact cell decomposition and approximate cell decomposition methods. Exact cell decomposition methods divide the configuration space into distinct cells. A connectivity graph is then constructed using these cells. Types of exact cell decomposition techniques include trapezoidal and triangular decomposition [19]. For trapezoidal decomposition, the first step is to obtain a polygonal region and place $\mathbf{q}_0$ and $\mathbf{q}_{goal}$ in the free space. Next, the free space is divided into trapezoids. Vertices are placed on every vertical line and at the centroid of each trapezoid. Lastly, the adjacency graph is searched for a path from $\mathbf{q}_0$ to $\mathbf{q}_{goal}$ [1]. Triangular decomposition is similar to trapezoidal decomposition, except this method divides C_{free} into triangles. Because of the difficulty associated with implementing exact cell decomposition methods, approximate cell decomposition methods are used more frequently since they estimate the configuration space as an occupancy grid map, which consists of square-shaped cells [2]. For binary occupancy grid maps, cells are either occupied or unoccupied. Cells that contain obstacles are considered occupied, while cells in the C_{free} space are unoccupied [2]. Approximate cell decomposition methods construct a graph using cell corners as vertices and connections between cell corners as edges. A grid search method can then be used to search the graph for a route from $\mathbf{q}_0$ to $\mathbf{q}_{goal}$.

Voronoi diagrams are geometric structures whose points have the same clearance from obstacles [28]. They have the benefit of ensuring that robots remain at a safe distance from obstacles, but they may generate longer paths because Voronoi diagrams have an affinity for remaining in C_{free} [1].

Combinatorial planning algorithms may have a fast convergence time, but they are impractical to implement because of their high complexity associated with dividing the free space into polygons [18]. In addition, they are unfeasible to implement for path planning problems with high degree of freedoms [30].

Sampling-based methods were developed to overcome shortcomings associated with combinatorial methods. These techniques employ collision detections to repeatedly search the configuration space for a path instead of characterizing the entire configuration space [19]. Sampling-based methods, such as RRT, have longer convergence times than combinatorial methods, but they are often used because they are easier to implement [16] [18], [27]. In addition, sampling-based methods will generate a path, if it exists, but the runtime of the algorithm may be infinite [4].

Categories of sampling-based path planning methods include probabilistic roadmaps (PRM) and rapidly exploring random trees (RRT) [31]. PRMs are sampling-based algorithms that do not build search trees, but begin the search by sampling random points $\mathbf{q}_{rand}$ in the C-space. These points are saved as milestones and are then connected to the neighboring milestones. Next, paths without collisions are saved as local paths, which are used by a graph search method to obtain a path [19]. By contrast, RRT algorithms begin by building a search tree from $\mathbf{q}_0$. They sample points and employ collision detections to add points to the search tree, until it reaches $\mathbf{q}_{goal}$ [17]. RRT algorithms have the advantage of generating a path without having to employ a graph search method.

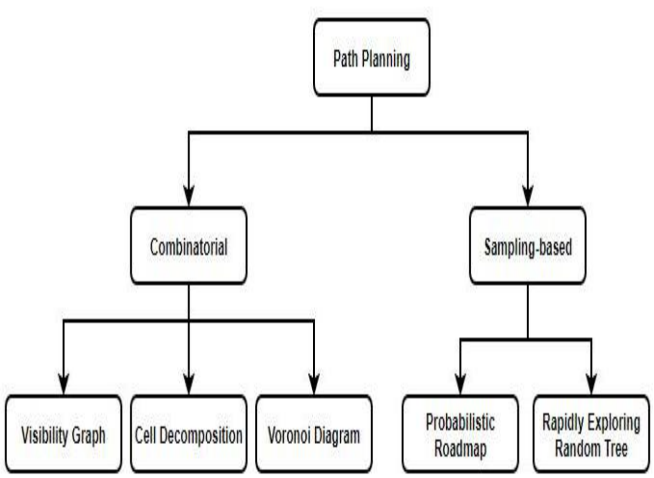


Figure 2. Classification of Path Planning Algorithms

The remainder of this paper is organized as follows: section 2 will review three types of rapidly exploring random tree algorithms to include the RRT, RRT*, and Informed RRT* algorithms. Section 3 will discuss simulations to evaluate their performance in two simulated environments, which are a L-shaped wall of obstacles and a wall of obstacles that has gaps in it. Section 4 will conclude with the highlights of these simulations. Lastly, Section 5 will discuss potential follow-on experiments.

2. RAPIDLY EXPLORING RANDOM TREE (RRT) ALGORITHMS

The Rapidly Exploring Random Tree (RRT) algorithm efficiently generates paths, while avoiding obstacles [13], [17]. This paper discusses three variations of this algorithm: the RRT, RRT*, and Informed RRT* algorithms.

2.1 Rapidly Exploring Random Tree (RRT)

The RRT algorithm creates a path from an initial configuration $\mathbf{q}_0$ to a goal configuration $\mathbf{q}_{goal}$. It then constructs a tree G, which consists of vertices V and edges E. The start location $\mathbf{q}_0$ and goal location $\mathbf{q}_{goal}$ are automatically added to the vertices list.

The tree begins at $\mathbf{q}_0$. It then randomly selects a point $\mathbf{q}_{rand}$ in the configuration space. If this point coincides with an obstacle, the algorithm selects a different random point. If the randomly generated point $\mathbf{q}_{rand}$ is in the free space, the algorithm determines the nearest point in the tree $\mathbf{q}_{nearest}$ to $\mathbf{q}_{rand}$.

This algorithm then attempts to draw a line from its nearest point $\mathbf{q}_{nearest}$ to the randomly generated point $\mathbf{q}_{rand}$. If this line collides with an obstacle, the algorithm selects a new random point. If the line does not collide with an obstacle, this line is drawn between the nearest point in the tree and the randomly generated point. The algorithm terminates when the tree arrives at the goal location $\mathbf{q}_{goal}$ [19]. This algorithm outputs tree G and Q_{sol}, which is the path from the start to the goal positions [17]. Table 1 contains the pseudocode for the RRT.

Table 1. RRT Algorithm Pseudocode

```
RRT(q_0, q_goal)
   V ← {q_0, q_goal}
   E ← Ø
   G = (V, E)
   Q_sol ← Ø
   q_rand ← q_0
   q_nearest ← q_0
   while GOAL_REACHED(q_rand, q_goal) == false
      q_rand ← RANDOM_CONFIG()
      if OBSTACLE_FREE(q_rand) == true
          q_nearest ← NEAREST(G, q_rand)
          if COLLISION_FREE(q_nearest, q_rand) == true
              G ← INSERT_NODE(q_nearest, q_rand, G)
              G ← EXTEND(q_nearest, q_rand)
          end
      end
   end
   Q_sol ← LIST_OF_COMPLETED PATHS(G)
   return G, Q_sol
end
```

2.2 RRT* Algorithm

The Rapidly Exploring Random Tree Star (RRT*) algorithm is an optimized RRT algorithm that incorporates near neighbor search and rewiring tree functions to reduce the distance traveled [22]. The RRT* algorithm has the asymptotic optimality property;

consequently, it converges to an optimal path, although it has a long convergence time [11], [14], [15].

This algorithm begins by initializing the search tree by adding the start configuration $\mathbf{q}_0$ and the goal configuration $\mathbf{q}_{goal}$ to the vertices list. It generates a random configuration $\mathbf{q}_{rand}$ in the free space, and the algorithm determines the nearest point $\mathbf{q}_{nearest}$ in the tree to $\mathbf{q}_{rand}$. If the path from $\mathbf{q}_{nearest}$ to $\mathbf{q}_{rand}$ is obstacle free, the near neighbor search function looks for nodes $\mathbf{q}_{near}$ that are within a radius k that is inputted to the function. The choose parent function then selects the optimal configuration from the nearby configurations $\mathbf{q}_{min}$, and it is added to tree G. The rewire function compares the distance traveled through $\mathbf{q}_{min}$ to the original distance to minimize the distance traveled [22]. This algorithm terminates when the number of searches N is reached. It then outputs tree G and Q_{sol} which is the path from the start to the goal positions. Table 2 contains the pseudocode for the RRT* algorithm.

Table 2. RRT* Algorithm Pseudocode

```
RRT*(q_0, q_goal, N, k)
V ← {q_0, q_goal}
E ← ∅
G = (V, E)
Q_sol ← ∅
q_rand ← q_0
q_nearest ← q_0
q_near ← q_0
q_min ← q_0
for i = 0 to N
    q_rand ← RANDOM_CONFIG()
    if OBSTACLE_FREE(q_rand) == true
        q_nearest ← NEAREST(G, q_rand)
        if COLLISION_FREE(q_nearest, q_rand) == true
            q_near ← NEAR(G, q_rand, k)
            q_min ← CHOOSE_PARENT(q_near, q_nearest, q_rand)
            G ← INSERT_NODE(q_min, q_rand, G)
            G ← REWIRE(G, q_near, q_min, q_rand)
        end
    end
end
Q_sol ← LIST_OF_COMPLETED_PATHS(G)
return G, Q_sol
end
```

2.3 Informed RRT* Algorithm

The Informed RRT* algorithm improves upon the RRT* algorithm by reducing the size of the search area, which results in shorter paths [5], [6]. The RRT and RRT* algorithms sample the entire search area, which cause their search trees to include nodes that are far away from the start and goal configurations. Paths that include nodes that are far away cause robots to travel long distances to get to their destinations. Instead, the Informed RRT* algorithm is most concerned with the area in close proximity to the start and goal locations. It samples the elliptical subset of the search area that has the start and goal locations as this ellipse's foci [7], [8]. Fig. 3 shows a diagram of this search ellipse that illustrates that c_{min} is the minimum distance from $\mathbf{q}_0$ to $\mathbf{q}_{goal}$. Table 3 contains the pseudocode for the Informed RRT* algorithm.

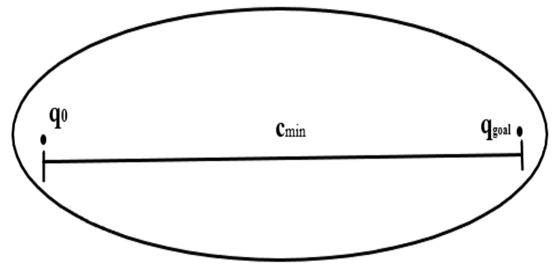


Figure 3. Search Ellipse for Informed RRT* Algorithm

Table 3. Informed RRT* Algorithm Pseudocode

```
Informed RRT*(q_0, q_goal, N, k)
V ← {q_0, q_goal}
E ← ∅
G = (V, E)
Q_sol ← ∅
q_rand ← q_0
q_nearest ← q_0
q_near ← q_0
q_min ← q_0
c_min ← || q_0 - q_goal ||
for i = 0 to N
    q_rand ← SAMPLE(q_0, q_goal, c_min)
    if OBSTACLE_FREE(q_rand) == true
        q_nearest ← NEAREST(G, q_rand)
        if COLLISION_FREE(q_nearest, q_rand) == true
            q_near ← NEAR(G, q_rand, k)
            q_min ← CHOOSE_PARENT(q_near, q_nearest, q_rand)
            G ← INSERT_NODE(q_min, q_rand, G)
            G ← REWIRE(G, q_near, q_min, q_rand)
        end
    end
end
Q_sol ← LIST_OF_COMPLETED_PATHS(G)
return G, Q_sol
end
```

3. SIMULATIONS

Two-Dimensional (2D) simulations for the RRT, RRT*, and Informed RRT* algorithms were conducted using the PythonRobotics toolbox [24], [25], [26].

Each algorithm's Python file was opened to initialize the experimental parameters. The maximum number of searches for each variation of RRT was set to 200. The boundaries of the search area were initialized by setting the maximum and minimum values of the x and y axes to -2 and 15 respectively. The start and goal nodes of the search trees were set to (0,0) and (10,10) respectively. In addition, two circular obstacle lists, as shown in Tables 4 and 5, were created so that the performance of each algorithm could be tested in two simulated environments. The first set of obstacles, shown in Figure 4, form an L shaped wall, while the second set of obstacles, shown in Figure 5, form a wall with narrow gaps between the obstacles.

Table 4. Obstacle List 1

X Coordinate of Center	Y Coordinate of Center	Radius
5	5	1
3	6	2
3	10	2
3	10	2
7	5	2
9	5	2
8	10	1
6	12	1

Table 5. Obstacle List 2

X Coordinate of Center	Y Coordinate of Center	Radius
5	5	0.5
9	6	1
7	5	1
1	5	1
3	6	1
7	9	1

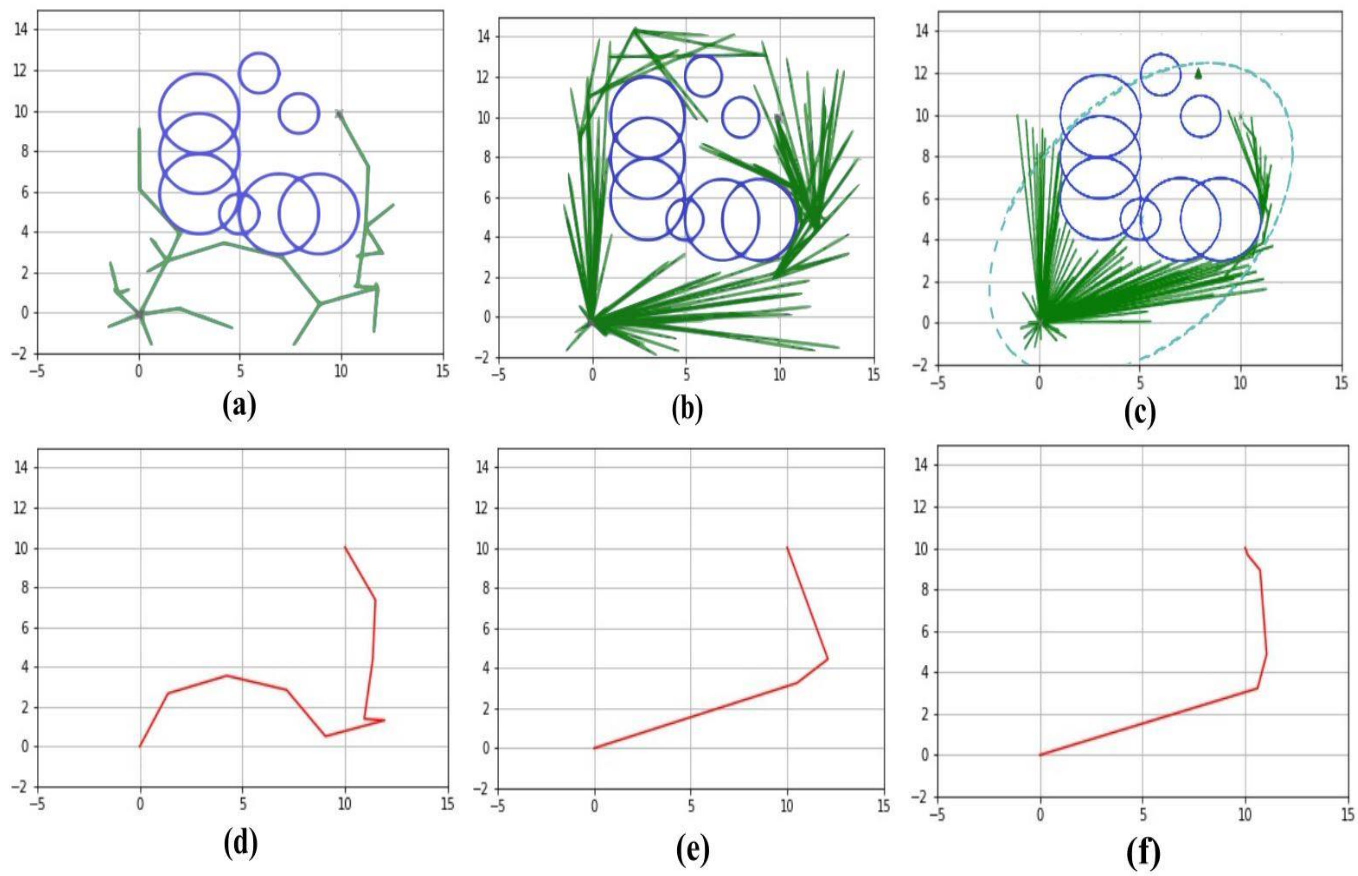


Figure 4. Simulations for Obstacle Set 1. Figures (a) and (b) correspond to Trial 2 for the RRT algorithm simulation. There are 11 nodes in path (b) because the second to the last node is located at (10.01, 9.97). The length of path (b) is 25.04. Figures (c) and (d) correspond to Trial 9 for the RRT* algorithm simulation. There are 5 nodes in path (d) because the second to the last node is located at (11.04, 7.02). The length of path (d) is 18.94. Figures (e) and (f) correspond to Trial 8 for the Informed RRT* algorithm. Figure (e) shows that the foci of the search ellipse are located at the start and goal locations. There are 6 nodes in path (f) because the second to the last node in this path is located at (10.13, 9.65), and the length of this path is 18.18.

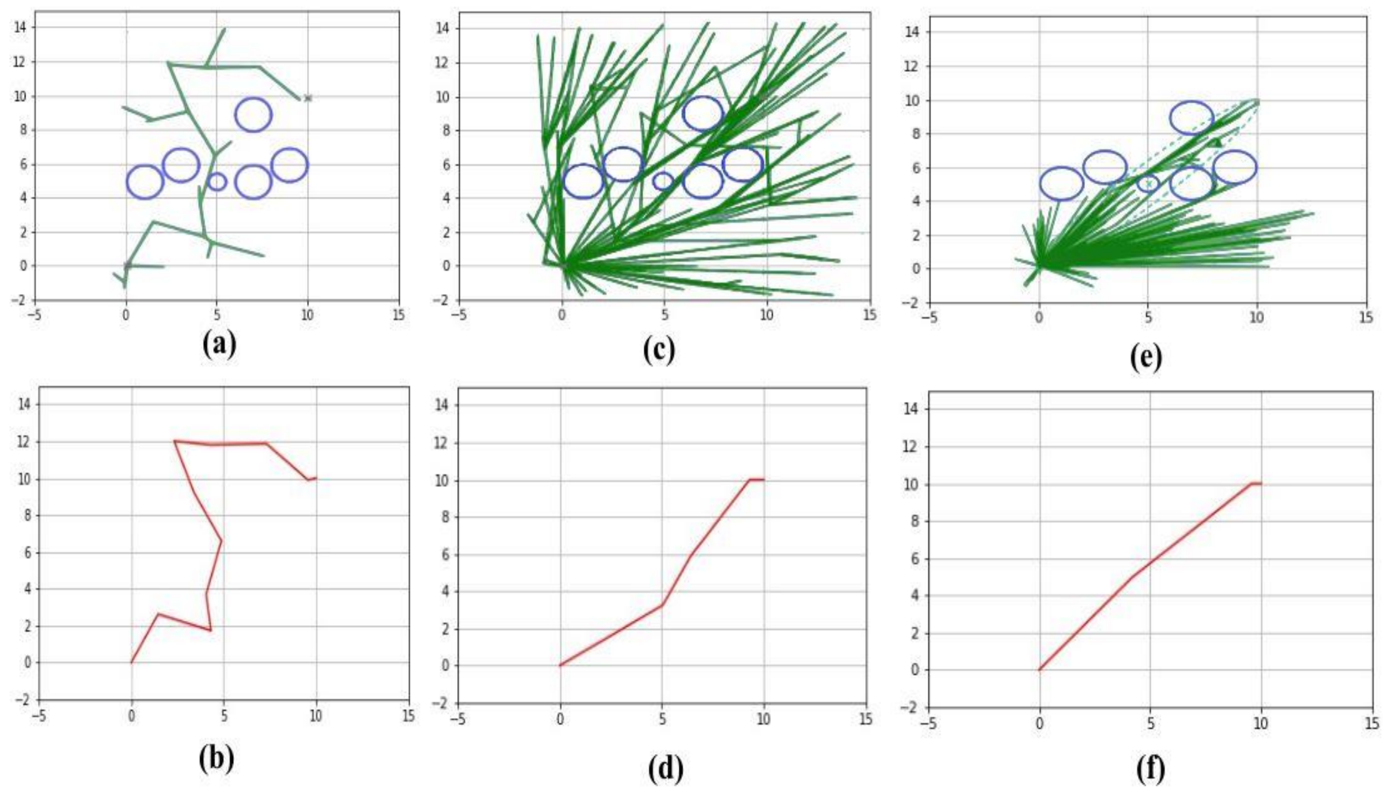


Figure 5. Simulations for Obstacle Set 2. Figures (a) and (b) correspond to Trial 5 for the RRT algorithm simulation. There are 11 nodes in path (b) because the second to the last node is located at (9.56, 9.88). The length of path (b) is 19.44. Figures (c) and (d) correspond to Trial 2 for the RRT* algorithm simulation. There are 5 nodes in path (d) because there are two nodes located at (9.31, 7.02) and (6.43, 5.91). The length of path (d) is 14.69. Figures (e) and (f) correspond to Trial 27 for the Informed RRT* algorithm. Figure (e) shows that the foci of the search ellipse are located at the start and goal locations. There are 4 nodes in path (f) because the second to the last node in this path is located at (9.58, 9.99), and the length of this path is 14.29.

The runtime, number of searches, and total number of nodes in the search tree were recorded for each simulation. If an algorithm successfully generated a path, the number of nodes in the path and the coordinates of nodes in this search path were also recorded. The data for these simulations were recorded in an Excel spreadsheet.

For these simulations, the distance between each set of consecutive points was calculated as

$$d_{i,i+1} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (1)$$

where x_i and y_i are the coordinates of the current point, and x_{i+1} and y_{i+1} are the coordinates of the next point in the path. The total length of each search path d_{tot} was computed as

$$d_{tot} = \sum_{i=1}^{n-1} d_{i,i+1} \quad (2)$$

where n is the total number of points in the path, and $d_{i,i+1}$ is the distance between each pair of consecutive points.

Each simulation was repeated for at least 30 trials or until a path could be found. The Central Limit Theorem states that the distribution of a sample approaches a Gaussian distribution as the number of samples increases [9].

The sample statistics for the runtime, number of searches, total number of nodes in the search tree, number of nodes in the path, and total path length were calculated and recorded in Tables 6, 7, 8, 9, 10, and 11 in the appendix. The runtime includes the amount of time for each variation of RRT algorithm to build a search tree and to generate the path from the start to goal locations. The total number of searches is equal to the total number of iterations for each algorithm. The number of searches for the RRT algorithm is less than 200 because this algorithm terminates as soon as a path from the start to the goal node is generated. By contrast, the number of searches for the RRT* and Informed RRT* algorithms are equal to 200 because these algorithms continue to improve the search path until the maximum number of searches is reached.

The number of nodes in the search tree include only the number of points that are successfully added to the search tree. Randomly selected points that are generated inside of the obstacle space are not added to the search tree. In addition, randomly generated points that are blocked by obstacles are not added to search trees. Furthermore, the number of nodes in the path is equal to number of points in the path from the start position to the goal position.

Sample statistics, to include the mean, median, mode, and standard deviation, are calculated to compare the performance of each algorithm [10].

The arithmetic mean is the average value of a quantity [10], [12]. The mean $\bar{x}$ is given as

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (3)$$

where x_i is a data value, and n is the total number of datapoints. The median is the middle value of the data set, the mode is of a set of datapoints is the value that occurs most often, and standard deviation is an indication of how far datapoints are spread apart [10] [12].

The standard deviation s is computed as

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2} \quad (4)$$

where n is the total number of samples, x is an individual sample, and $\bar{x}$ is the mean.

The data for the first obstacle set is provided in Tables 6, 7, and 8, and that of the second obstacle set is provided in Tables 9, 10, and 11. These data tables are found in the appendix section. For the first obstacle set, the runtimes for the RRT, RRT*, and Informed RRT* were 0.51s, 0.47s, and 0.54s with standard deviations of 0.26s, 0.26s, and 0.31s respectively. For the second obstacle set, the runtimes for the RRT, RRT*, and Informed RRT* were 0.53s, 0.45s, and 0.47s with standard deviations of 0.30s, 0.23s, and 0.24s respectively. Algorithms with the shorter runtimes are more efficient than those with longer runtimes. For obstacle set 1, the mean path lengths for the RRT, RRT*, and Informed RRT* algorithms were 23.76, 19.07, and 18.19 with corresponding standard deviations of 2.93, 0.57, and 0.25 respectively. For this set of obstacles, the Informed RRT* algorithm found the shortest path lengths, and its path length is the most predictable because its standard deviation is the smallest. For the second obstacle set, the mean pathlengths for the RRT, RRT*, and Informed RRT* algorithms were 19.60, 14.59, and 14.28 with corresponding standard deviations of 3.07, 0.37, and 0.15 respectively. The Informed RRT* algorithm found the shortest path lengths, and its path length was the most predictable because its standard deviation is the smallest. Algorithms with shorter path lengths enables the robot to get to its goal location faster than algorithms with longer path lengths. The mean path lengths for obstacle set 1 were longer than that of obstacle set 2 because the path planning algorithms traveled through gaps in the obstacles, as illustrated in Figure 5. For the Informed RRT* algorithm, the success rate for obstacle set 1 was lower than that of obstacle set 2 because the algorithm had difficulty finding a path around the L-shaped block of obstacles. Consequently, the configuration of the obstacles impacts the runtime and path length, regardless of the type of path planning algorithms. However, for obstacle sets 1 and 2, the Informed RRT* algorithm had a success rate of only 41.67% and 94% respectively, in finding a path. By contrast, for both sets of obstacles, the RRT and RRT* algorithms had a success rate of 100% in finding a path.

4. CONCLUSIONS

Path planning algorithms are sampling-based robot navigation techniques that determine a robot's route from its beginning to its goal configurations. This paper reviewed three variations of RRT algorithms: RRT, RRT*, and Informed RRT* algorithms. Simulations were conducted to compare their performances. The success rates for finding a path was 100% for the RRT and RRT* algorithms because the robot was able to reach the goal locations. These algorithms are also reliable for both obstacle sets. The RRT algorithm is easier to implement because it returns a path as soon as one is found; however, it had the longest path lengths. Since the RRT* is a refined version of RRT algorithm, it has the capabilities of the RRT algorithm with the added benefit of shorter path lengths. The Informed RRT* algorithm was more complicated to implement, yielded the shortest path lengths, but its success rate of 41.67% and 94% for obstacle sets 1 and 2 respectively was much lower than that of the RRT and RRT* algorithms. The results of this study show that the arrangement of the obstacles impact the effectiveness of these algorithms in path planning.

5. FUTURE WORK

Future experiments involve implementing and repeating the simulations with the variable step-size RRT algorithm, which adapts the distance that the search tree extends based on the proximity of obstacles [32]. The number of nodes in the search tree and number of nodes in the final path for the variable step-size RRT algorithm simulation will be compared to those from the RRT, RRT*, and Informed RRT* algorithm simulations. In addition, the simulations that were conducted are applicable to plan paths for robots that travel in a plane, such as autonomous ground vehicles [21]. Future research would modify these algorithms to operate in three-dimensions (3-D), so that they could be applied to robots, such as unmanned aerial vehicles [20], [23]. To conduct these simulations, spherical landmarks would be created. The distance calculations in each path planning algorithm will also be modified to include x, y, and z coordinates.

6. ACKNOWLEDGMENTS

A hearty thank you to the Electrical Engineering Department and to the Engineering Dean's Fellows Program at the FAMU-FSU College of Engineering for their support.

7. REFERENCES

[1] K. Arras. 2014. Robot Motion Planning. (February 2014). Retrieved April 26, 2020 from http://srl.informatik.uni-freiburg.de/teachingdir/ws13/slides/12-RobotMotionPlanning.pdf

[2] S. Bhowmick, J. Mukhopadhyay, and A. Kanti Deb. 2017. Fast Path planning on planar occupancy grid exploiting geometry of obstacles. *2017 Ninth International Conference on Advances in Pattern Recognition (ICAPR)* (2017). DOI:http://dx.doi.org/10.1109/icapr.2017.8593059

[3] J. V. D. Brand and D. Nanongkai. 2019. Dynamic Approximate Shortest Paths and Beyond: Subquadratic and Worst-Case Update Time. *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)* (November 2019), 436–455. DOI:http://dx.doi.org/10.1109/focs.2019.00035

[4] M. Elbanhawi and M. Simic, 2014. Sampling-Based Robot Motion Planning: A Review. *IEEE Access* 2 (February 2014), 56–77. DOI:http://dx.doi.org/10.1109/access.2014.2302442

[5] J. D. Gammell. 2014. Informed RRT*. (March 2014). Retrieved March 2, 2020 from https://robotic-esp.com/code/informed-rrtstar/

[6] J. D. Gammell. 2014. Informed RRT* @ UTIAS (IROS 2014). (July 2014). Retrieved February 20, 2020 from https://www.youtube.com/watch?v=nsl-5MZfwu4

[7] J. D. Gammell, Timothy D. Barfoot, and Siddhartha S. Srinivasa. 2018. Informed Sampling for Asymptotically Optimal Path Planning. *IEEE Transactions on Robotics* 34, 4 (2018),966–984. DOI:http://dx.doi.org/10.1109/tro.2018.2830331

[8] J. D. Gammell, Siddhartha S. Srinivasa, and Timothy D. Barfoot. 2014. Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2014), 2997–3004. DOI:http://dx.doi.org/10.1109/iros.2014.6942976

[9] A. Ganti. 2020. What Is the Central Limit Theorem (CLT)? (January 2020). Retrieved February 26, 2020 from https://www.investopedia.com/terms/c/central_limit_theorem.asp

[10] L. Green. 2008. Mean, Mode, Median, and Standard Deviation. (2008). Retrieved February 26, 2020 from http://www.ltcconline.net/greenl/courses/201/descstat/mean.htm

[11] B. C. Guevara, "An Overview of the Class of Rapidly-Exploring Random Trees," M.S. thesis, Dept. Humanities., Utrecht Univ., Utrecht, Netherlands, 2018.

[12] B. L. Illowsky and S. L. Dean. 2018. *Introductory Statistics*, Houston, TX: OpenStax, Rice University.

[13] F. Islam, J. Nasir, U. Malik, Y. Ayaz, and O. Hasan. 2012. RRT∗-Smart: Rapid Convergence Implementation of RRT∗ Towards Optimal Solution. *2012 IEEE International Conference on Mechatronics and Automation* (2012). DOI:http://dx.doi.org/10.1109/icma.2012.6284384

[14] S. Karaman, and E. Frazzoli, 2011. Sampling-based Algorithms for Optimal Motion Planning. *The International Journal of Robotics Research* 30, 7 (2011), 846–894. DOI:http://dx.doi.org/10.1177/0278364911406761

[15] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, 2011. Anytime Motion Planning using the RRT*. *2011 IEEE International Conference on Robotics and Automation* (June 2011). DOI:http://dx.doi.org/10.1109/icra.2011.5980479

[16] M. C. Kim and J. B. Song. 2017. Informed RRT* with improved converging rate by adopting wrapping procedure. *Intelligent Service Robotics* 11, 1 (February 2017), 53–60. DOI:http://dx.doi.org/10.1007/s11370-017-0242-9

[17] S. M. LaValle. 1998. Rapidly-Exploring Random Trees: A New Tool for Path Planning. (October 1998).

[18] S. M. LaValle. 2006. Planning Algorithms. Cambridge University Press, New York, NY, USA.

[19] S. M. LaValle, 2011. Motion Planning: The Essentials. *IEEE Robotics & Automation Magazine* 18, 1 (March 2011), 79–89. DOI:http://dx.doi.org/10.1109/mra.2011.940276

[20] M. Li, Q. Sun, and M. Zhu. 2019. UAV 3-Dimension Flight Path Planning Based on Improved Rapidly-exploring Random Tree. *2019 Chinese Control and Decision Conference (CCDC)* (2019). DOI:http://dx.doi.org/10.1109/ccdc.2019.8832832

[21] B. W. Millar. Multi-Resolution Obstacle Mapping with Rapidly-Exploring Random Tree Path Planning for Unmanned Air Vehicles (2011). *Theses and Dissertations*. 2620. https://scholarsarchive.byu.edu/etd/2620

[22] I. Noreen, A. Khan and Z. Habib. 2016. A Comparison of RRT, RRT*, and RRT*-Smart Path Planning Algorithms. *IJCSNS International Journal of Computer Science and Network Security* 16, 10 (October 2016). DOI:http://dx.doi.org/10.1109/icma.2012.6284384

[23] M. Ozcan and M. M. Ankarali. 2019. Feedback Motion Planning for a Dynamic Car Model via Random Sequential Composition. *2019 IEEE International Conference on Systems, Man and Cybernetics (SMC)* (2019). DOI:http://dx.doi.org/10.1109/smc.2019.8913917

[24] A. Sakai. 2020. AtsushiSakai/PythonRobotics. (February 2020). Retrieved February 20, 2020 from https://github.com/AtsushiSakai/PythonRobotics

[25] A. Sakai. 2018. Welcome to PythonRobotics's documentation! (2018). Retrieved February 20, 2020 from https://pythonrobotics.readthedocs.io/en/latest/index.html#

[26] A. Sakai, Daniel Ingram, Joseph Dinius, Karan Chawla, Antonin Raffin and Alexis Paques. (2018). PythonRobotics: a Python code collection of robotics algorithms. *ArXiv, abs/1808.10703*.

[27] O. Salzman. 2019. Sampling-Based Robot Motion Planning. Commun. ACM 62, 10 (September 2019), 54–63. DOI:https://doi-org.proxy.lib.fsu.edu/10.1145/3318164

[28] D. Schmitt and K. Vyatkina. 2012. On Voronoi Diagram in the Line Space and their Generalizations. *2012 Ninth International Symposium on Voronoi Diagrams in Science and Engineering* (2012), 122–125. DOI:http://dx.doi.org/10.1109/isvd.2012.22

[29] O. Soussi, R. Benatitallah, D. Duvivier, A. Artiba, N. Belanger, and P. Feyzeau, In *5th IESM Conference*, (Rabat, MA), Institute of Electrical and Electronics Engineers.

[30] L. Toma. 2017. Combinatorial Motion Planning (2017). Retrieved April 26, 2020 from http://www.bowdoin.edu/~ltoma/teaching/cs3250-CompGeom/spring17/Lectures/cg-motionplanning1.pdf

[31] J. Velagic, D. Delimustafic, and D. Osmankovic. 2014. Mobile robot navigation system based on Probabilistic Road Map (PRM) with Halton sampling of configuration space. *2014 IEEE 23rd International Symposium on Industrial Electronics (ISIE)* (2014), 1227–1232. DOI:http://dx.doi.org/10.1109/isie.2014.6864789

[32] Y. Wang and Y. Huang. 2019. Mobile Robot Path Planning Algorithm Based on Rapidly-Exploring Random Tree. *2019 IEEE International Conferences on Ubiquitous Computing & Communications (IUCC) and Data Science and Computational Intelligence (DSCI) and Smart Computing, Networking and Services (SmartCNS)* (2019). DOI:http://dx.doi.org/10.1109/iucc/dsci/smartcns.2019.0017

8. APPENDIX

The data for the RRT, RRT* and Informed RRT* algorithms corresponding to obstacle list 1 and 2 are provided in Tables 6, 7, 8, 9, 10 and 11 respectively.

Table 6. RRT Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.51	0.55	0.06	0.95	N/A	0.26
Number of Searches	42.63	37	10	146	37	25.88
Total Number of Nodes in Tree	24.73	21	10	97	28	16.52
Number of Nodes in Path	9.97	10	8	13	10	1.19
Path Length	23.76	22.81	19.88	30.59	N/A	2.93

Table 7. RRT* Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.47	0.47	0.06	0.97	N/A	0.26
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	149	149	125	166	149	11.41
Number of Nodes in Path	4.87	5	4	6	5	0.43
Path Length	19.07	18.98	18.09	20.73	N/A	0.57

Table 8. Informed RRT* Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.54	0.60	0.02	1.00	N/A	0.31
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	111.99	112.50	80	150	100	15.95
Number of Nodes in Path	6.20	6	5	8	6	0.85
Path Length	18.19	18.09	17.86	18.82	N/A	0.25

Table 9. RRT Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.53	0.58	0.003	0.94	N/A	0.30
Number of Searches	20.87	21	6	53	21	9.48
Total Number of Nodes in Tree	15.33	13	6	48	13	8.32
Number of Nodes in Path	8.20	8	6	12	8	1.49
Path Length	19.60	19.27	14.32	27.95	N/A	3.07

Table 10. RRT* Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.45	0.48	0.03	0.89	N/A	0.23
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	183.60	184	175	191	185	4.23
Number of Nodes in Path	4.23	4	4	5	4	0.43
Path Length	14.59	14.51	14.21	15.9	N/A	0.37

Table 11. Informed RRT* Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.47	0.52	0.05	0.97	N/A	0.24
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	171.75	176.50	133	187	178	13.38
Number of Nodes in Path	4.6	4.5	4	6	4	0.67
Path Length	14.28	14.24	14.19	14.98	N/A	0.15

A Review of Paper #96

Roussev, Roussi
Splunk
rroussev@splunk.com

Estimated Utility of:
"A Review of Rapidly Exploring Random Tree Path Planning Algorithms"
Utility to Community: $15000
Utility to Society: $15000

Overall, this is an interesting survey paper covering tree path planning algorithms from different papers. I enjoyed how it brought them under the same notation and compared them.

Here is a list of things I would improve:

1. While understanding that space is limited, I would have enjoyed a more thorough description of the combination methods, a comparison against the sampling-based ones, and a more informative taxonomy (e.g. Figure 2).

2. A few nits on the pseudocode: In "Table 1. RRT", Qrand is not initialized before the loop. There are different definitions of seemingly similar operations: OBSTACLE_FREE(qrand), OBSTACLE_FREE(qrand, qnearest) and COLLISION_FREE(qnearest, qrand). Maybe clarify/refactor the code mostly for readability (the text description seems clear enough). Same with RRT*(q0,qgoal), G ← RRT*(q0, N) and G ← RRT*(G, qrand). You can probably also remove the call to LIST_OF_COMPLETED PATHS() since RRT* returns it.

3. It will be interesting to perform experiments with a wider variety of obstacles and starts/goals. There are two paragraphs that dwell a bit too much on the contents of tables 6-8/9-11 and are pretty much the same except for the numbers plugged in. Maybe explore ways of sharing the data, so it doesn't take valuable space in the paper.

4. For further discussion, it might be useful to cover the tradeoffs of each algorithm, how to potentially overcome their limitations (especially for Informed RRT*) and when it is best to use each.

Response to Mr. Roussev's Comments

Jermyn, Jacqueline
Florida State University
`jmj15c@my.fsu.edu`

Answer To Review:
"A Review of Paper #96"
Utility Review: $15000

A Review of Rapidly Exploring Random Tree Path Planning Algorithms

Ribeiro, Eraldo
Florida Institute of Technology
eribeiro@fit.edu

Estimated Utility of:
"A Review of Rapidly Exploring Random Tree Path Planning Algorithms"
Utility to Community: $1
Utility to Society: $1

The paper is a survey on a specific type of path-planning algorithms, namely Rapidly Exploring Random. The three methods compared in paper are: RRT, Rapidly Exploring Random Tree Star (RRT*), and Informed RRT* algorithms. The methods are well described by algorithm listing. Good simulations are provided in the experimental sections. The paper is well written.

However, I think some of the plots could be improved. For example, Figure 4(e) and Figure 5(e) have a different scale from the other plots shown in the same figures.

Finally, a survey paper is expected to provide future directions that are more specific than the ones suggested in this paper.

Suggested future directions seem to be simply to modify algorithms but do not provide any specific direction, citation, or path to address issues.

Response to Dr. Ribeiro's Comments

Jermyn, Jacqueline
Florida State University
`jmj15c@my.fsu.edu`

Answer To Review:
"A Review of Rapidly Exploring Random Tree Path Planning Algorithms"
Utility Review: $1000

Part II

Track Main

Vision-based decision system for humanoid robot

Bartosz Bok
Warsaw University of Technology
Faculty of Electronics
and Information Technology
bok.bartosz@gmail.com

Dominik Górczyński
Warsaw University of Technology
Faculty of Power
and Aeronautical Engineering
gorczyn1892@gmail.com

Kacper Mikołajczyk
Warsaw University of Technology
Faculty of Power
and Aeronautical Engineering
kacper.mikolajczyk96@gmail.com

Maksymilian Szumowski
Warsaw University of Technology
Institute of Aeronautics
and Applied Mechanics
mszumowski@meil.pw.edu.pl

ABSTRACT

This article presents the method of using a vision system based on a camera, which is the main source of information about the surroundings of a humanoid robot. Basing on the image analysis, the robot is able to make a decision about its next moves. The decision system based on the camera image is shown on the example of sumo fighting humanoid robots, competition that become increasingly popular in the world. This article describes the use of Hough transform on the camera image, which allows to determine the position of the robot on the ring. Then, relying on the calculation of the largest gradient tangent to the line of the ring, robot can find the position of the opponent. Thanks to this, the decision making system is able to calculate the estimated trajectory based on the opponent's movement history, which allows the robot to approach opponent as soon as possible. The above-mentioned method has been implemented on real humanoid. The article contains the results of a robot test in the ring, which is used in competitions.

Keywords: vision system, decision system, humanoid robot.

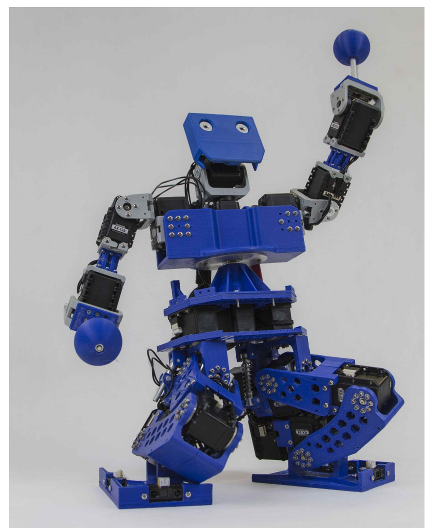


Figure 1. Melson, a humanoid robot designed and built by members of Student's Robotics Associacion at Warsaw University of Technology

1. Introduction

Technological development has recently changed the reality radically. People often use technology to make their lives easier. Robots will increasingly accompany them in private and professional life. They will have to move in a space where people also move, so there is a problem because many of them move thanks to the wheel drive. This is a good way only on a flat surface. However, the space in which people move is adapted for bipeds. It has a lot of flat surface, as well as many places that can be obstacles for robots, such as stairs, curbs or other uneven terrain. Therefore, a good solution may be the development of robots that move like humans, i.e. humanoids. Recently, the topic of humanoid robots is becoming more and more popular. The world of science is trying to popularize solutions of bipedal robots. There are many competitions in which such robots can compete with each other, such as races, matches or robot battles. Many robotics are involved in the development of algorithms for controlling robots [1], [2]. In order to make the right decisions, the robot needs to get information about its surroundings. For this purpose, most robots have different types of sensors, e.g. distances (ultrasonic, laser, etc.), color. However, when there is information about distance or color, robot do not get information about what object it is or where it can be in space. The LIDAR sensor gives information about the distances of objects on the plane around the robot, however also no major information about objects is obtained, e.g. what is the color distribution in space [3].

By using the camera as the main source of information about the environment and appropriate analysis of its image, robot is able to know the characteristics of the space around the robot much more accurately. It is useful when planning the robot's paths of travel [4] or various types of humanoid robot jobs [5].

2. Humanoid Robot Melson

In the Students' Robot Association at the Warsaw University of Technology, a humanoid robot has been prepared with a developed autonomous operation. The robot is called 'Melson', which has a height of 50 cm and a weight of 2.6 kg. He has 19 degrees of freedom, distance sensors and a camera on his head. The robot is au-

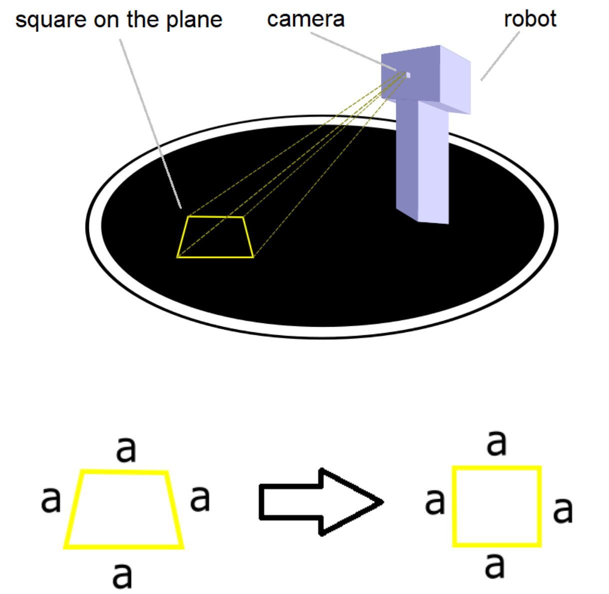


Figure 2. Transformation of the camera view into a plane parallel to the ring

tonomous, and all calculations are performed on a Raspberry Pi microcontroller or IntelNUC computer, which is located on its back. Melson took part in the international robot contests in Europe. It got silver medal in the humanoid sumo competition in Robot Challenge in Bucharest. The humanoid won also a lot of competition in Poland. However, attempts are made to improve this project, which will allow him to get more prizes.

3. Vision System

The usage of the vision system in robotics is becoming more and more popular. Thanks to proper analysis of the image provided by the camera, the robot is able to recognize its surroundings. This solution can be used in humanoid robot battles. For a robot on the battlefield, knowledge of its surroundings is important. It is practically impossible to use ordinary sensors, e.g. distances, because they can only return information about the location of other objects directly in front of the sensor. However, to know the location of the robot in the ring, it is important to be able to recognize the end lines of this field. Dohyō ring parameters are given before the fight, therefore, knowing its dimensions and colors, the vision system can be prepared before. A good way is to use a vision system that recognizes its position based on the image.

3.1 Hough Transform

There is a need for the robot to find a circle on the border of the battlefield - dohyō. This can help it to not get out of the ring and possibly find an opponent. A good solution for finding straight lines or circles in an image is to use the Hough Transform method. However, it requires a view of the ring from a direction perpendicular to it. It is therefore necessary to initially transform the image received from the camera, because the lines seen from the perspective of the robot's head are read as ellipses. However, having data

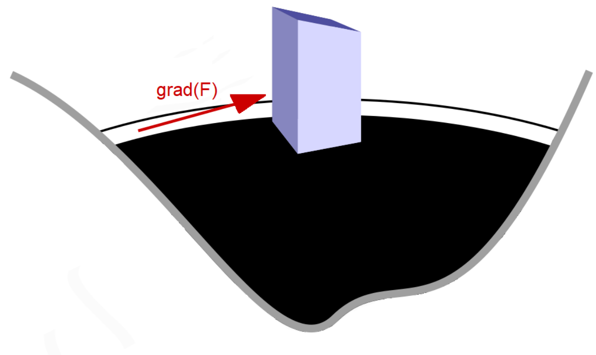


Figure 3. A color gradient tangent to the line surrounding the ring, where $grad(F)$ is gradient of F function representing pixel intensity

as to the position of the camera relative to the ground, the computer can project this image to the plane of the ring.

The camera is located on the head of the robot at a height of 25cm and is at an angle of 30 degrees to the plane parallel to the ring. Suppose there is a virtual square with a side length of 'a' on the plane of the ring. Then from the camera perspective it will be seen as a trapezoid. Inverse transformation can be made to project the entire image received from the camera onto the dohyō plane.

Once there is the ring line represented as a circle, it can be found using the Hough Transform method. It should be noted that the radius of the ring R is given, because in various competitions the radius of the ring is 80 cm. The first stage in Hough Transform is the initial separation of the outline of the objects in the image. The Laplacian function can be used on pixel intensity values. This will allow to extract the outlines of the objects in the image, because the lines around dohyō are white and the ring inside is black.

The equation describing a circle:

$$(x-x_0)^2+(y-y_0)^2=R^2, \tag{1}$$

where (x_0, y_0) are coordinates of the circle center, (x, y) are coordinates of points on the circle and R is given radius of the circle. Coordinates of a circle (x0, y0) are sought.

For each separated pixel, the so-called processed image creates the so-called Hough space. This is done by drawing circles with a radius R and the center of each extracted pixel, and the point where the most circles intersect in this space is most likely the center of the circle in the original image.

3.2 Location recognizing

By projecting the camera view onto the ring plane, it is able to find out the location of the robot. The image created after the transformation will be in the shape of a trapezoid, which can be interpreted as a view perpendicular to the axis of the cone of camera visibility. The tip of this cone is in the camera.

After casting the view on the ring and recognizing the opponent, the computer can create a map representing the robot's surroundings.

3.3 Identifying the opponent

The line recognition system can also be used to recognize the position of the opponent in the ring. Assuming that the robot will see the lines in the camera image, the opponent will most likely be

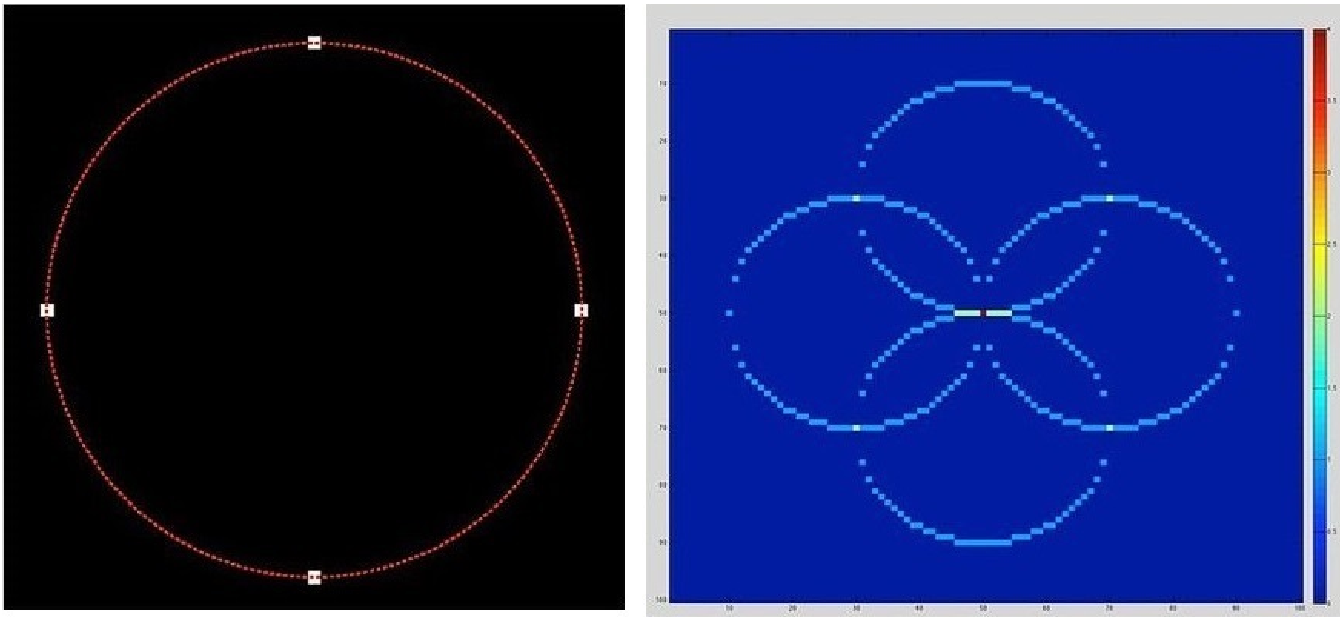


Figure 4. Finding a circle by the Hough transform method

where the line is broken. A good approach is to apply a tangent gradient to the line surrounding the circle. The function of the gradient of the tangent line to the circle is sought whose largest change is at the point where the ring line breaks.

After finding the white circle marking the end line of the ring, it is assumed that the opponent will break this line in the image. Therefore, the largest gradient is tangent to the circle line, and in the place where the gradient is the largest it is assumed that the opponent is located.

Assuming that the pixel intensity function is marked with white color F, its gradient is marked with the symbol grad (F).

Finding the opponent by the method of breaking the ring line will not get its exact position in the ring, but only the surroundings of its position. The place where the opponent may be located is illustrated in figure 3. Therefore, after finding the place where the line surrounding the ring is broken, a distance sensor attached to the servo is used, which rotates towards the arc, where the opponent is located. The servomechanism is attached to the abdomen of the robot. Only this sensor will provide accurate information about the opponent's distance to Melson.

4. Decision Making

The robot is able to execute some of the previously generated and saved trajectories. These trajectories allow it to do simple moves, for example: step forward or backward, rotation(15,30,45,60 or 90 degrees) or punching an enemy. The goal of the decision system is to choose moves, which helps the robot approach the opponent at minimum time. At the beginning of the battle the robot needs to find its location in the area. Basic assumption is that the global coordinate system is fixed at the center of the battlefield. After taking the first photo, the program places the Global Coordinate System (GCS) inside the circle. The first thing to do is to use the Hough Transform method to calculate a position of the protagonist and get the initial position of the robot.

After every step done by Melson, the program updates the position and orientation. It knows the previous position of the robot and distance passed, so computer can easy calculate the new one using vector of displacement. After that robot can find a position of the opponent in the GCS. During the first iteration robot calculate an angle of rotation needed for standing opposite to the enemy. In the second step the robot find opponent's new position. It allows Melson to calculate the vector of its antagonist's displacement. Next, robot can also predict the line which it will probably follow.

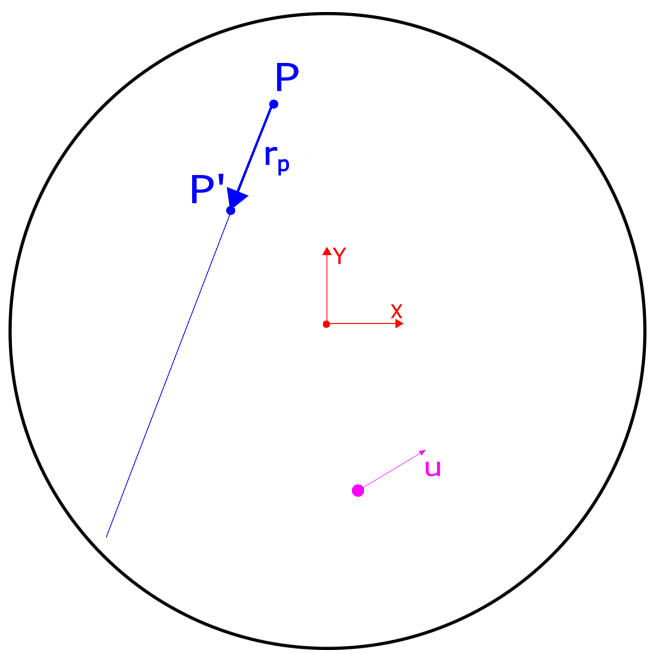


Figure 5. Finding the opponent. Vector r shows its displacement

When knowing the equation of that line it is easy to get a distance between Melson and that line (vector q). Also known is distance to the enemy (vector p). Using scalar product it is possible to calculate an angle between these vectors (β).

$$\beta = arccos(\frac{p \circ q}{|p||q|}) \tag{2}$$

The next step is to calculate an angle of rotation ϕ. After rotation unit vector u' should be collinear with the bisector of the angle ß. Then the robot moves one step forward.

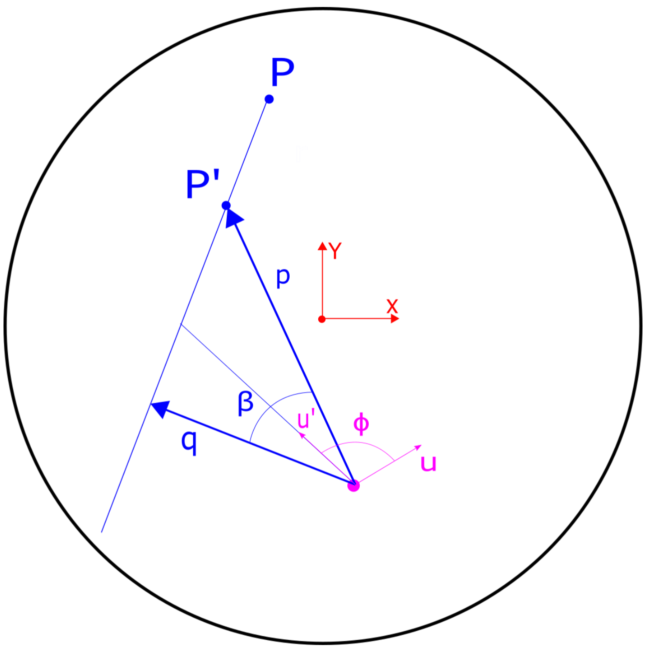


Figure 6. Angle of rotation ϕ

Melson repeats these operations. During every iteration it updates its position and opponent's walking line. When the robot stands less than 30 cm from the goal, it goes straight forward to approach the enemy and punch it. That method has two cases. First: the opponent is placed opposite to the robot. Knowing the enemy's behavior, the algorithm tries to reduce the distance between the two robots. Second: the opponent is placed backward and running away. Then the robot pursues it. Finally, Melson can punch the opponent in its back, which can be more effective.

5. Experiments

The experiment was decided to be divided into 3 stages:

1. Finding the end line of the ring and determining Melson's position on it.
2. Finding an opponent and determining its trajectory.
3. Making a decision about Melson's move relative to its opponent.

5.1 Experiment 1

At this stage of the experiment, the ring detection algorithm was examined based on individual photos taken with a camera placed on Melson's head. Tested on the basis of two photos shown of fig. 8:

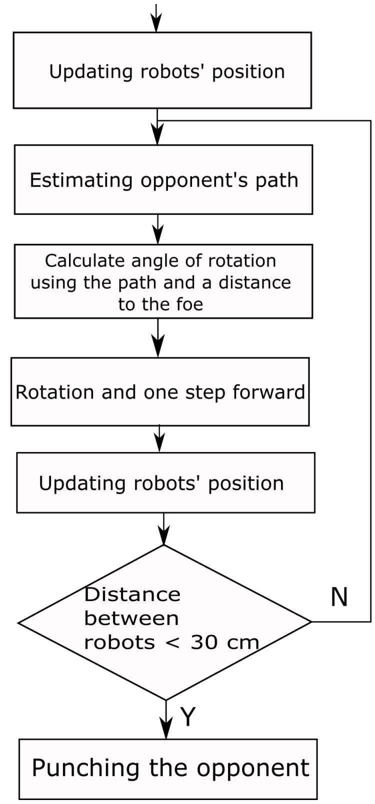


Figure 7. Block diagram of the decision system

As a result of the transformation of the photo into a view perpendicular to the ring and using the Hough transformation to find the best-fit circle, the photos presented in figure 8 were obtained shown together with raw photos. The found circle is marked in green, the center of the circle as a red point, and the blue lines are the camera's visibility. The circles indicate more or less the good location of the real circle. The relative approximation error is approximately 3%.

The above transformation of the photo into a view parallel to the ring indicates the border of the image cone that the camera sees. The camera lens is located at the intersection of two lines. On this basis, Melson can find its location:

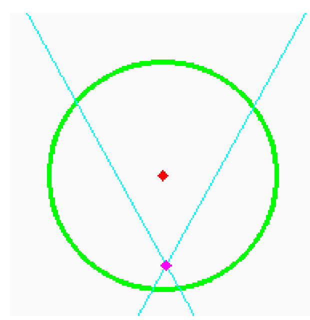


Figure 9. Calculation of the robot's position based on the photo of the ring

Melson's position is marked in pink. The true location of the humanoid also differs from the predicted one by about 3%. The results are satisfactory.

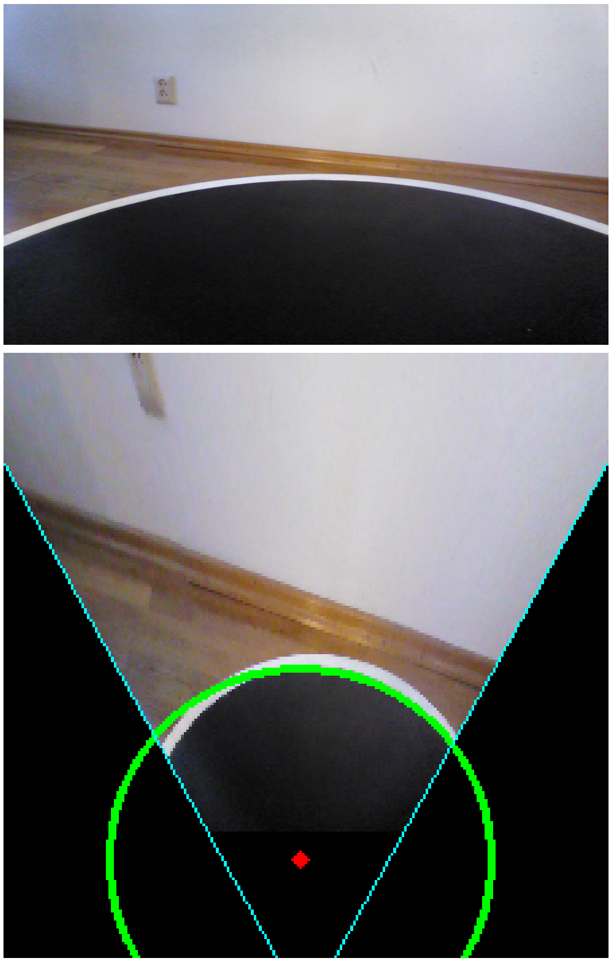


Figure 8. Transformation of the photo into a view parallel to the ring

5.2 Experiment 2

During the experiment, a small plastic model of the KUKA robot was used as the opponent which Melson is looking for in the ring. For this purpose, the tangent gradient method to the line surrounding the ring is used, described in the previous chapter.

At this stage of the experiment, it is assumed that Melson is stationary and his opponent moves. Every now and then (approximately the time for Melson to take one step) a picture of the surroundings is taken and the opponent is sought. In the case of this experiment, it was decided to examine its motion based on three photos taken every 3 seconds.

As shown in the figure 10, the robot moves to the right and each time crossed the white arc surrounding the ring. These images were obtained after the Hough transformation.

In figure 10 the robot was found in a good surroundings of its true location. The circle found was marked in green, the center of the circle was marked as a red point, the light blue lines were the camera visibility, and the blue point was the expected location of the opponent. The relative approximation error is 3%.

5.3 Experiment 3

To plan a path the robot needs two different positions of the opponent. Using that information the program extrapolates the line

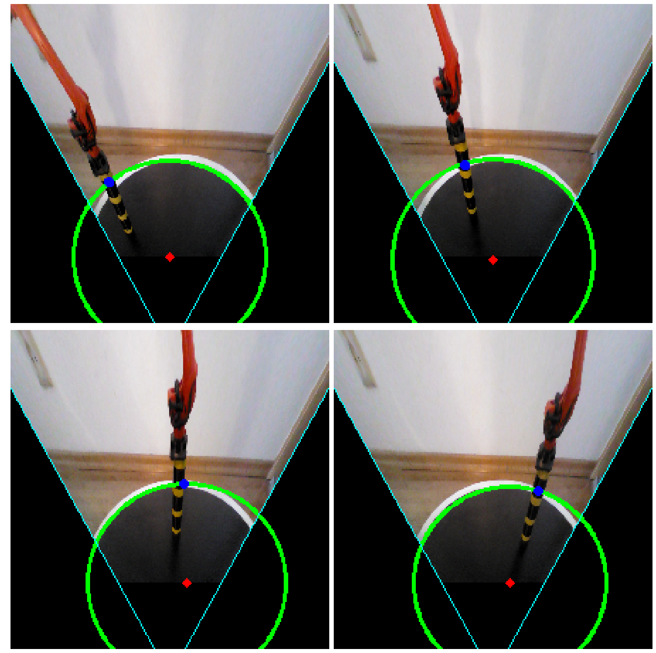


Figure 10. Calculation of the opponent's moving position

followed by the KUKA model. After estimating the equation of the line the robot calculates the distance to it. The error of that operation is approximately 6%. The program also calculates a vector between the robot and the opponent using vectors of the robots' positions. The calculated length of that vector differs from the real one by 3%.

The last step is to calculate an angle the robot needs to turn as it was shown in the "Decision making" paragraph. The Melson cannot turn a custom angle. It has to choose one from this list: 15,30,45,60 or 90 degrees. The algorithm executes the one, which is the nearest to the desired angle ϕ. In this example the angle was 12° clockwise, so the robot rotated through 15°.

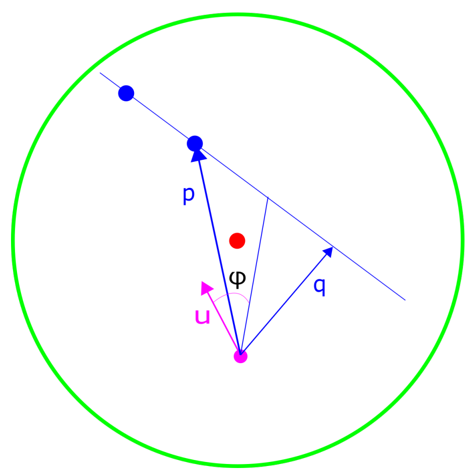


Figure 11. Result of the experiment. The robot should rotate clockwise through the angle ϕ

6. Conclusions

Humanoids are becoming an increasingly popular topic in robotics. More and more scientists, students and engineers deal with them, so it is important to develop an appropriate system of their operation. Using the camera and appropriate image analysis the robot can get a lot of information about its environment. A good example is the Hough transformation, which can help recognize shapes, e.g. circles, as shown in this article. After obtaining information about the space around the humanoid, the robot can go to the decision-making system. It will help the robot behave adequately to the dynamically changing area, as illustrated by the example of sumo competition.

Acknowledgments

Authors are members of Students' Robotics Association (Koło Naukowe Robotyków) at the Faculty of Power and Aeronautical Engineering, Warsaw University of Technology. Research was done cooperating with the Faculty of Power and Aeronautical Engineering within the Framework of "Najlepsi z Najlepszych 4.0!" program.

References

[1] C. R. Torrico, A. B. Leal, and A. T. Watanabe, "Modeling and supervisory control of mobile robots: A case of a sumo robot," *IFAC-PapersOnLine*, vol. 49, no. 32, pp. 240 – 245, 2016. Cyber-Physical Human-Systems CPHS 2016.

[2] H. Erdem, "Application of neuro-fuzzy controller for sumo robot control," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9752 – 9760, 2011.

[3] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," July 2014.

[4] K. Okada, M. Inaba, and H. Inoue, "Walking navigation system of humanoid robot using stereo vision based floor recognition and path planning with multi-layered body image," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 3, pp. 2155–2160 vol.3, 2003.

[5] C. O. Vilão, D. H. Perico, I. J. Silva, T. P. D. Homem, F. Tonidandel, and R. A. C. Bianchi, "A single camera vision system for a humanoid robot," in *2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol*, pp. 181–186, 2014.

The paper about geometric visual localisation and opponent detection for robotic sumo

Barták, Roman
Charles University
bartak@ktiml.mff.cuni.cz

Estimated Utility of:
"Vision-based decision system for humanoid robot"
Utility to Community: $5
Utility to Society: $1

The authors present a system for visual localisation of a robot in sumo arena, finding location of the opponent, and deciding where to move. First, it should be said that there is nothing specific about humanoid robots in this paper. The presented techniques are completely independent of the robot, so highlighting humanoid (bi-pedal) robots in the introduction, in conclusions and even in the title is misleading. The suggested method is described well at high-level, but with no technical details.

First, about the localisation, which is based on detecting the border of sumo arena. How do you know the parameters for transformation as shown in Figure 2? Is there any calibration before the game starts? Or are you using parameters like diameter of arena, height and angle of the camera as mentioned in the paper? Aren't these parameters changing as the robot moves? Specifically for humanoid robots, the location of camera with respect to surface is changing a lot and even a small change in degree may cause big inaccuracy. About the location found, you do not really know the exact location - the robot can be anywhere on the circle. Look at Figure 9, how do you know that the robot is looking north? What if it is looking south or in whatever direction? You have no such information. It is probably not needed for the robotic sumo, but you must be clear about it in the text.

About finding the oponent, you can use you method only when the robot sees the opponent. What does the robot do if the opponent is behind? No decision about it is in your decision system. Also, you only very briefly mentioned that the distance to opponent is measured by distance sensor, but nothing more is said about it. Is the robot actively moving to get a better position for observation, like if the opponent is not visible? Is decision making taking on account behaviour of the opponent? For example, you calculate speed of opponent, is this used to predict where the opponent will be so the robot can move to that location rather than to the current location? Actually, the decision system is very reactive with very simple strategy used (for example, no escape if the opponent is attacking). The experiments are described in very high-level too, no details and no measurements are given so it is hard to evaluate the contribution.

In summary, this is a nice exercise on geometric reasoning in robotics. Robustness of the method is a bit unclear and the decision system is still very simple with no real model of opponent.

Vision-based decision system for humanoid robot

Bok, Bartosz
Warsaw University of Technology
bartos.bok@gmail.com

Answer To Review:
"The paper about geometric visual localisation and opponent detection for robotic sumo"
Utility Review: $5

The presented robot is prepared for humanoid sumo competitions, hence the need to develop a system for recognizing the opponent and making the right decision. The rules of the humanoid sumo competition are most often universal around the world (e.g. ring dimensions, line thickness, etc.). This is why the article is based on the specificity of these battles. An important aspect is also that the camera must be on the head of a humanoid, which should be taken into account when building a given system. As mentioned earlier, the parameters of the ring are known before the competition in the regulations. Humanoid takes a photo to analyze every step it takes when it is motionless, which is why the camera's height and angle of inclination is constant and known thanks to the joint configuration. The transformation presented in Figure 2 can be made, using the previously loaded algorithm. There is a need for knowledge about robots location relative to the center of the ring and about the position of the opponent relative to the robot. It does not matter in which geographical direction the robot looks, only the relative position to the opponent is important. Based on the known initial position of the humanoid, knowing its kinematics, movement distances and angles of rotation, its exact location can be counted at any moment. When the battle begins, robots are placed opposite to each other. There is small chance, that the opponent can move behind our robot. But it is still possible. In this case, the robot rotates through 90 degrees, then tries to find the enemy. If it does not work, it keeps rotating until it finally find the opponent. The proximity sensor is placed on a servo shaft, so it is able to rotate. Anaylizing an image from the camera provides the angular position of the foe. Then the sensor approaches that angle and measures the distance. During a battle a referee can give a penalty to a robot which behaves passively. There is a risk that escaping can be interprated as a passive action. The robot always seeks a confrontation with its enemy. Humanoid robots are relatively slow. The decision system does not aim the current position of the foe, because it continously moves. It means that robot pursues the enemy, which takes a time. Predicition of the future state of the robot allows to get the opponent in shorter time. Due to recent events related to the pandemic, access to the laboratory was difficult to carry out a solid experiment with the humanoid opponent. To this end, we decided to replace the opponent with the toy model we moved.

Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation

Alexander Frye[1], Hadi Fekrmandi [2], Randy C. Hoover[1], and Amir Hossein Tamjidi[3]

[1]Department of Computer Science and Engineering, South Dakota School of Mines & Technology, 501 E. Saint Joseph St., Rapid City, SD 57701, Email: Alexander.Frye@mines.sdsmt.edu, Randy.Hoover@sdsmt.edu
[2]Department of Mechanical Engineering, South Dakota School of Mines & Technology, 501 E. Saint Joseph St., Rapid City, SD 57701, Email: Hadi.Fekrmandi@sdsmt.edu
[3]Software Engineer, Driving Algorithms Team, Chess Dr, Foster City, California 94404, Email: ahtamjidi@gmail.com

ABSTRACT

The objective of the Swarm Demonstration for Space Autonomous Tasks (SD-SAT) project is to demonstrate proximity formation flying of a swarm of CubeSats using vision-based relative camera sensors. In this paper, a linear model of orbital dynamics and a state-feedback controller were implemented for proximity formation flying in low-earth orbit satellite simulations. The model was tested and proven to accurately represent the nonlinear orbital dynamics equations. The controller was able to maneuver a group of three cube-satellites into the desired formation. The method is shown to be adaptable and scalable for N number of satellites in proximity operation missions. Development and integration of the proposed method into the vision-based guidance, control, and navigation subsystems will pave the way for new attitude determination and control technology for swarm proximity formation flying missions.

1. INTRODUCTION

According to NASA's Small Spacecraft Technology Program (SSTP), a spacecraft is called a "small spacecraft" when its mass is below 180 kg. CubeSats are a class of small spacecrafts called nanosatellites weighting in the range of 1-10 kg. CubeSats are built to standard dimensions known as Units (i.e. 1U, 2U, 3U, or 6U) where each unit is a 10 cm x 10 cm x 10 cm cube [1]. CubeSats are suitable for formation flying missions because of their modular nature and low cost [2]. Multiple satellites working together can offer additional capabilities over monolithic satellites due to their cooperative behavior within the swarm. Theses advantages include increased flexibility, accuracy, maximum range, significantly enhanced flexibility (adaptability, scalability, evolvability, and maintainability), and robustness (reliability, survivability, and fault tolerance) [3]. The capabilities needed to support swarm missions go beyond operator-specified geometry, alignment, or separation, but also include cross-link communication while maintaining position in the formation [4]. Table 1 shows a few planned scientific missions by a swarm of CubeSats.

In many scientific missions, particularly when accurate spatial and temporal data must be provided, accurate state estimation is required to satisfy the mission objectives. For example, simultaneous measurements from a 3-dimensional volume of space could be made feasible by a swarm and these measurements are advantageous for a variety of studies in space physics and Earth science.

Table 1. Scientific research missions by swarm of CubeSats missions

Mission Name	Mission Objective
CPOD - CubeSat Proximity Operations Demonstration [5]	Poximity operations and docking of two 3U CubeSats
Frisbee – A Platform for Small Satellite Science Swarms [6]	Space Weather Advanced Research Mission (SWARM)
Olfar - Swarm Telescope Constellation [7]	A Radio Telescope Based on Nano-Satellites in Moon Orbit
EDSN - Edison Demonstration of SmallSat Networks [8]	Intersatellite Communications Architecture
Nodes - Networked spacecraft command and control [9]	synchronized, multipoint scientific measurements
HelioSwarm - Swarm Mission Design in High Altitude Orbit [10]	Studying Physics of Space Plasma Turbulence
CLICK - CubeSat Laser Infrared CrosslinK Mission [11]	Optical Communications Cross-link communication

Swarm Orbital Dynamics Advisor (SODA), is a platform developed by NASA that takes general configuration commands and provides the orbital maneuvers needed to achieve the desired formation flying through swarm relative motion (shown in Figure 1).

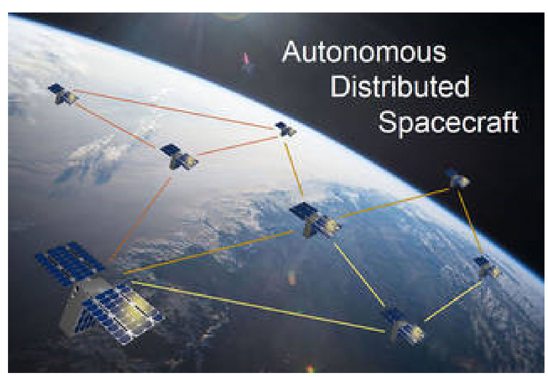


Figure 1. Swarm Orbital Dynamics Advisor (SODA) controls the configuration of satellites in a distributed swarm mission [12] (Figure courtesy of NASA Ames Research Center).

Although there has been promising progress in recent years in the field of swarms of CubeSats, the current state of the art of guidance navigation and control (GNC) relies particularly on the Global Positioning System (GPS). The performance of the GPS subsystem was evaluated in a mission to study space weather measurements during the Radio Aurora explorer II (RAX-2) CubeSat mission [13]. The results presented a the GPS-derived error of 4.02 m for position and 0.48 m/s for velocity.

While the GPS subsystem may have sufficient accuracy for providing useful positioning information for some small satellite swarm missions (for example weather data measurements from distributed points of the atmosphere), many scientific missions require much higher position accuracy. Examples include those involving multi-satellite docking, earth and space imaging, and inspection missions around larger satellites. One of the fundamental drawbacks of current GNC techniques is their reliance on a centralized process using global positioning system (GPS) [14]. While real-time positioning computed by standard GPS service is adequate for some dispersed applications (i.e. constellation missions), inherent position discontinuities are not acceptable for proximate formation flying missions demanding high-precision science instruments for simultaneous measurements [15]. In addition, deep space swarm missions in GPS-denied environments will rely on alternative orbit determination using radio or optical-based measurements [12]. Optical techniques such as Smart Phone Video Guidance (SVGS) [16] have been considered for obtaining an accurate relative state estimation of the members of the swarm and offer an efficient relative state estimation in close proximity. Prior work using visual-based systems for accurate state/pose estimation has shown promising results to Unmanned Ground Vehicles (UGVs) operating in GPS denied environments [17]. Use of an illuminated fiducial system (referred to as an AR or Augmented Reality Tag was studied for proximity formation flying [18, 19]). In this method, a unique identifier is allocated to each CubeSat within the swarm and it is shown that the vision-based pose estimator provides relative spacial pose estimation that is highly accurate compared to GPS. In the current paper, a computationally efficient controller is proposed for CubeSat proximity formation flying missions. It is assumed that the relative states are available through both GPS and vision-based methods. The challenge is for agents to maintain certain relative formation and recover the disturbances imposed due to the presence of low density atmosphere in low earth orbit (LEO). The proposed system provides a solution for maneuvering CubeSats into the desired formation in LEO. The proposed method of control may also be developed further to completely rely on relative measurements for deep space applications where accurate positioning of CubeSats is important for multiple-node sensor measurements and mapping.

2. GUIDANCE, NAVIGATION and CONTROL

To control cube-satellites in orbit around earth, a linearized model and controller are required. To test the linear model and controller, an accurate, nonlinear model for orbital dynamics was implemented. This nonlinear model is a frictionless, single-body model of gravitational acceleration. These limitations shouldn't affect the simulation, as in a low number of orbits the loss in velocity from air friction is negligible, and as long as the satellites are not near the moon there are no other bodies that would significantly alter their trajectories.

The acceleration due to gravity from any body to another is

$$a_g = \frac{Gm_1}{r^2}\frac{\vec{r}}{r}, \text{ or } a_g = \frac{\mu\vec{r}}{r^3},$$

where G is the gravitational constant ($6.673x10^{-11}$ Nm^2/kg^2), m_1 is the mass of the orbited body (for earth, $5.972x10^{24}$ kg), μ is the standard gravitational parameter (for earth, $3.986x10^{14}$ m^3/s^2), r is the distance in meters between the center of mass of each body, and $\vec{r}$ is the vector from the satellite to the center of mass of the orbited body. This model is accurate, but not entirely useful because as a spacecraft orbits, the radius vector $\vec{r}$ is constantly changing. To solve this, the equation can be re-written Cartesian coordinate system in terms of the $[x, y, z]$, where the radius can be redefined as

$$\vec{r} = \vec{x} + \vec{y} + \vec{z} \text{ and } r = \sqrt{x^2 + y^2 + z^2}.$$

Now that there is a nonlinear reference model, candidate linear models can be tested. The linear model used (Equation 1) is based on the model presented in [20], with small changes to the matrix D_0 because that model did not work as intended. 0_3 is a 3 by 3 matrix of zeros, I_3 is the dimension 3 identity matrix, and μ and r are body and orbit-specific constants defined above. Changing μ affects what body the model pertains to, and changing r changes the radius of orbit it is linearized about.

$$\begin{aligned} A_0 &= \begin{bmatrix} 0_3 & I_3 \\ D_0\mu/r^3 & S_0\sqrt{\mu/r^3} \end{bmatrix} & B_0 &= \begin{bmatrix} 0_3 \\ I_3 \end{bmatrix} \\ D_0 &= \begin{bmatrix} -3 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{bmatrix} & S_0 &= \begin{bmatrix} 0 & 2 & 0 \\ -2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{aligned} \tag{1}$$

This model is linearized about an equatorial, circular orbit, but as long as the orbital plane remains constant. If a polar orbit (an orbit not in the plane of the equator) was required, the axes could be rotated to bring that orbit into the $[x, y]$. The state vector for this model is given by

$$\mathbf{x} = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z}]^T,$$

which comprises the position and velocity data in $[x, y, z]$ but contains no rotation or angular velocity data. This model is for a single satellite, and said satellite's velocity and acceleration can be modeled with $\dot{\mathbf{x}} = A_0\mathbf{x} + B_0\mathbf{u}$. To validate the usage of a linearized version of the nonlinear dynamics for the purpose of swarm control, the researchers simulated each model with identical initial conditions and plotted the result. This test involves plotting several satellites in orbits around earth, with realistic orbit radius and velocity. When an uncontrolled satellite ($\mathbf{u} = 0$) is plotted in a circular orbit of radius $6,478$ km (100 km above the earth's surface) with an orbital velocity of $7,848$ m/s (the velocity of a circular orbit at this height), the linear model perfectly recreates the nonlinear dynamics, even with small initial positional offsets from the ideal orbit (Figure 2).

To create a model for multiple satellites in a swarm, the A_0 and B_0 matrices from Equation 1 can be combined, with the assumption that the dynamics of each satellite are completely independent from one another (this holds as long as the satellites do not thrust directly at one another while in close proximity).

The state matrices A and B are then

$$A = I_n \otimes A_0,$$

$$B = I_n \otimes B_0,$$

and $\mathbf{x}$ is then a vector of n x_0 blocks, for example if $n = 3$

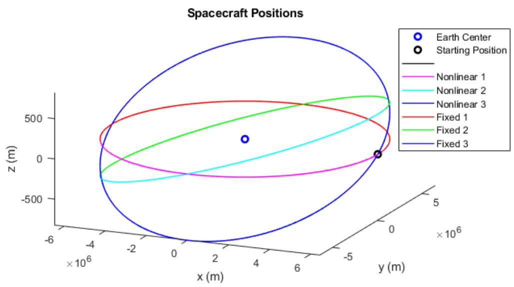


Figure 2. Performance of the linearized vs. nonlinear models in three different orbits

$$\mathbf{x} = [x_0 \;\; x_0 \;\; x_0]^T.$$

In these equations, $\otimes$ is the Kronecker product and I_n is the identity matrix of dimension n. For a 3-satellite system: $\mathbf{x}$, A, and B can be created using the previous equations where $n = 3$. In block matrix form, with 0_n being an $n \times n$ matrix of zeros and $0_{n,m}$ being an $n \times m$ matrix of zeros, $\mathbf{x}$, A, and B are

$$\mathbf{x} = \begin{bmatrix} x_0 \\ x_0 \\ x_0 \end{bmatrix} \quad A = \begin{bmatrix} A_0 & 0_6 & 0_6 \\ 0_6 & A_0 & 0_6 \\ 0_6 & 0_6 & A_0 \end{bmatrix} \quad B = \begin{bmatrix} B_0 & 0_{6,3} & 0_{6,3} \\ 0_{6,3} & B_0 & 0_{6,3} \\ 0_{6,3} & 0_{6,3} & B_0 \end{bmatrix}. \tag{2}$$

From this model, a state feedback controller was designed to stabilize the relative positions of these satellites in close proximity to each other both in terms of position and velocity. This controller is based off of a PD controller, with the main goal of being fuel efficient and having a low thrust magnitude. Adding an integral term could potentially reduce the steady-state position error, but tends to add oscillations and could potentially destabilize the system. The full state equation is now: $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} = A\mathbf{x} + BK(\mathbf{x} - g)$ where g comprises the desired positions of satellites 2 and 3 relative to satellite 1, and K is the gain matrix. The state feedback equation K is described in Equation 3, where $0_{n,m}$ is an $n \times m$ matrix of zeros and $k_p = -0.000001$ and $k_d = -0.001$. These gains are extremely small because the maximum thrust of a cube-satellite is very limited, especially when using high-efficiency propulsion mechanisms such as ion thrusters.

$$K_0 = \begin{bmatrix} k_p I_3 & k_d I_3 \end{bmatrix} \quad K = \begin{bmatrix} 0_{9,6} & 0_{9,6} & 0_{9,6} \\ -K_0 & K_0 & 0_{3,6} \\ 0_{3,6} & 0_{3,6} & 0_{3,6} \\ -K_0 & 0_{3,6} & K_0 \end{bmatrix}. \tag{3}$$

3. RESULTS

To demonstrate this controller, numerous tests were conducted with a 3-satellite formation. Every test had one satellite (hereafter referenced as satellite 1) that started orbiting in a circular 100 km above earth, with the other two satellites (satellites 2 and 3) starting in near-identical orbits, just 100 meters away from satellite 1 in the X, Y, and Z directions. It should be noted that the 5 points on each plot specify the plot location at each increment of 1000 seconds. At 0 seconds the satellites are at point 1, at 1000 seconds the satellites are at point 2, and so on. This is done to give an indication of time in the plots without a time axis.

In one specific example, satellites 2 and 3 had initial and desired positions detailed in Table 2. This example was selected because it shows that the controller is capable of moving the satellites with very little thrust from a loose to a tight formation. The satellites are controlled to hold their positions relative to the global coordinate system (holding their positions relative to a distant observation target such as a planet of star).

Table 2. Positions in meters relative to satellite 1

	X	Y	Z
Satellite 2 start	100	0	0
Satellite 3 start	-100	0	100
Satellite 2 desired	-10	-10	0
Satellite 3 desired	-10	10	0

First, the uncontrolled system was plotted. The satellite positions plotted around the earth (Figure 3) shows them orbiting, but because of the incredible orbit radius there is no indication of their separation. In contrast, when the positions of satellites 2 and 3 are plotted relative to satellite 1 (Figure 4) they form an odd pattern. This can be better understood when looking at each axis overlaid, and plotted vs. time, as they are in Figure 5. It should be noted that a 100 km orbit of earth has an orbit period of 5,300 seconds (just under one and a half hours). In these figures the satellites oscillate from their start position to the direct opposite position relative to satellite 1 and back every orbit. They don't stay in the same formation, but neither do they spread out away from each other.

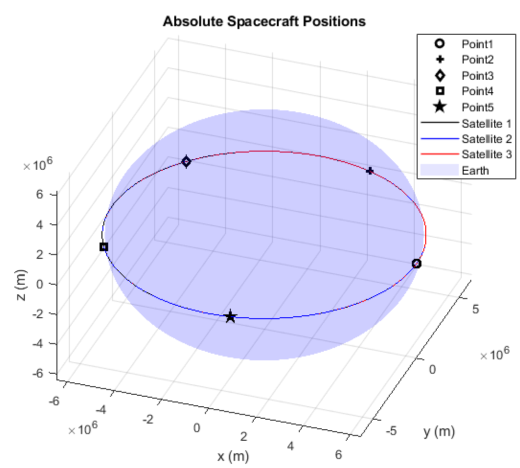


Figure 3. Absolute positions of satellites 1, 2, and 3 (uncontrolled).

In contrast to the uncontrolled test, the controlled plots have no oscillatory behavior. The plot of the absolute positions (Figure 6) of each satellite is very similar to the uncontrolled plot, again because of the extreme distances involved. In contrast, the positions of satellites 2 and 3 relative to satellite 1 are controlled to slowly travel to the goal positions (Figure 7). This same phenomena can also be seen in Figure 8.

4. CONCLUSION

The swarm of the small satellites are capable of delivering substantial science return, at a portion of the cost of larger counterparts. The proximity formation flying of swarm of CubeSats is particularly challenging because of the limitations in propulsion, power and

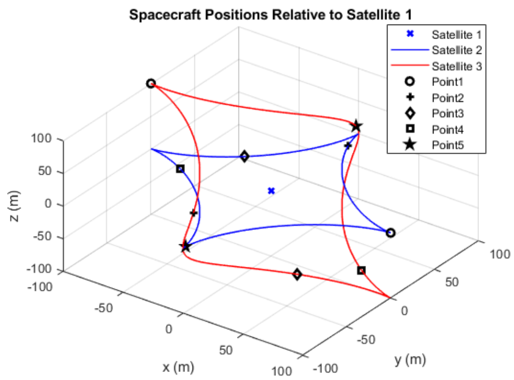


Figure 4. Positions of satellites 2 and 3 relative to satellite 1 (uncontrolled).

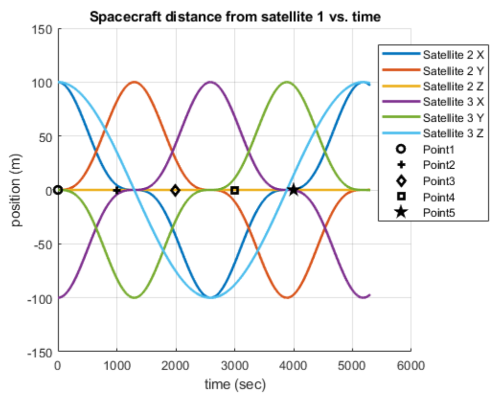


Figure 5. Positions of satellites 2 and 3 relative to satellite 1 (uncontrolled).

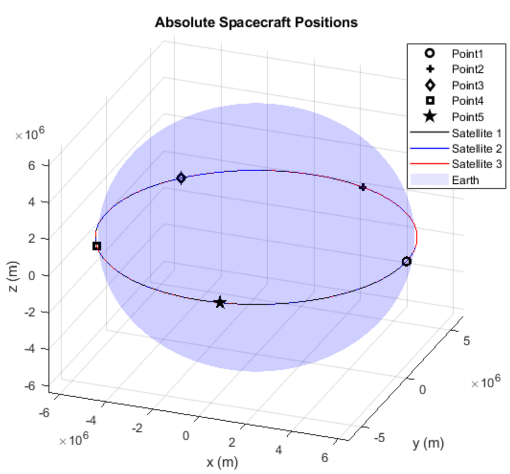


Figure 6. Absolute positions of satellites 1, 2, and 3 (controlled)

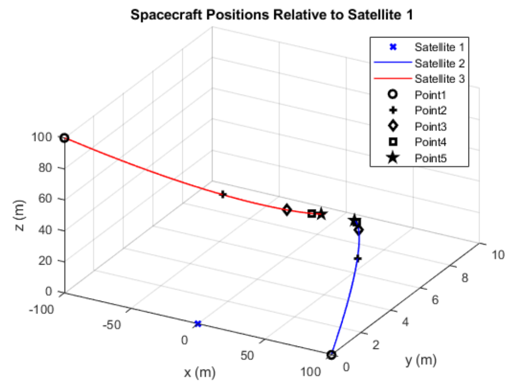


Figure 7. Positions of satellites 2 and 3 relative to satellite 1 (controlled)

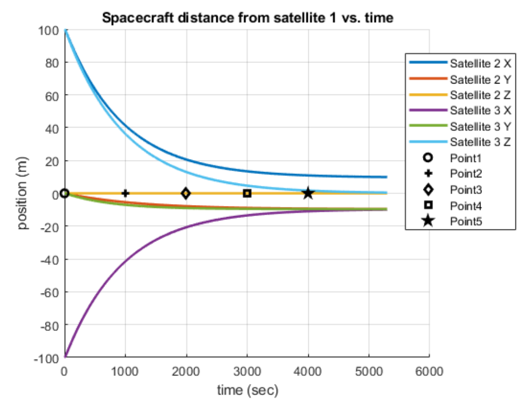


Figure 8. Positions of satellites 2 and 3 relative to satellite 1 (controlled)

computational capabilities of small spacecrafts due to their small size. In this work, these shortcomings are addressed by development of an accurate and robust guidance, navigation and control subsystem. The proposed method of this study using the linear model of orbital dynamics and a low cost state-feedback controller for swarm has shown promising performance. This will enable the agents of swarm work collaboratively to deliver the high quality scientific measurement. Moreover, for missions beyond the low earth orbit, two key technical challenges for realizing spacecraft swarms in GPS-denied environments of deep space that has been tackled include: (1) fast and stable inter-connectivity among the spacecraft using only relative state measurement and (2) adequate state estimation to accurately and continuously determine the exact position of the ships forming the swarm.

5. ACKNOWLEDGMENT

Authors would like to express gratitude to Dr. Edward Duke, director of South Dakota NASA EPSCoR program for funding this research under Research Initiation Grant (RIG). Also, we greatly acknowledge both John Rakoczy and Brian Bae from control systems design and analysis division of NASA Marshall Space Flight Center for their countless feedback and support of this project.

References

[1] J. Crusan and C. Galica, "Nasa's cubesat launch initiative: Enabling broad access to space," *Acta Astronautica*, vol. 157, pp. 51–60, 2019.

[2] F. Y. Hadaegh, S.-J. Chung, and H. M. Manohara, "On development of 100-gram-class spacecraft for swarm applications," *IEEE Systems Journal*, vol. 10, no. 2, pp. 673–684, 2016.

[3] S. Bandyopadhyay, G. P. Subramanian, R. Foust, D. Morgan, S.-J. Chung, and F. Hadaegh, "A review of impending small satellite formation flying missions," in *53rd AIAA Aerospace Sciences Meeting*, p. 1623, 2015.

[4] J. Cockrell, R. Alena, D. Mayer, H. Sanchez, T. Luzod, B. Yost, and D. Klumpar, "Edsn: A large swarm of advanced yet very affordable, cots-based nanosats that enable multipoint physics and open source apps," 2012.

[5] C. W. Roscoe, J. J. Westphal, and E. Mosleh, "Overview and gnc design of the cubesat proximity operations demonstration (cpod) mission," *Acta Astronautica*, vol. 153, pp. 410–421, 2018.

[6] A. da Silva Curiel, M. Meerman, D. Liddle, S. Schwartz, and M. Sweeting, "Frisbee–a platform for a small satellite science swarms," 2003.

[7] S. Engelen, C. J. Verhoeven, and M. J. Bentum, "Olfar, a radio telescope based on nano-satellites in moon orbit," 2010.

[8] J. Hanson, J. Chartres, H. Sanchez, and K. Oyadomari, "The edsn intersatellite communications architecture," 2014.

[9] J. Hanson, A. G. Luna, R. DeRosee, K. Oyadomari, J. Wolfe, W. Attai, and C. Prical, "Nodes: A flight demonstration of networked spacecraft command and control," 2016.

[10] L. Plice, A. Dono Perez, and S. West, "Helioswarm: Swarm mission design in high altitude orbit for heliophysics," 2019.

[11] K. Cahoy, P. Grenfell, A. Crews, M. Long, P. Serra, A. Nguyen, R. Fitzgerald, C. Haughwout, R. Diez, A. Aguilar, *et al.*, "The cubesat laser infrared crosslink mission (click)," in *International Conference on Space Optics—ICSO 2018*, vol. 11180, p. 111800Y, International Society for Optics and Photonics, 2019.

[12] T. Conn, A. Perez, L. Plice, and M. Ho, "Operating small sat swarms as a single entity: Introducing soda," 2017.

[13] J. Arlas and S. Spangelo, "Gps results for the radio aurora explorer ii cubesat mission," in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 123, 2013.

[14] L. Winternitz, B. Bamford, S. Price, A. Long, M. Farahmand, and R. Carpenter, "Gps navigation above 76,000 km for the mms mission," tech. rep., NASA Goddard Space Flight Center, Greenbelt, MD, United States, 2016.

[15] A. Richardson, J. Strom, and E. Olson, "AprilCal: Assisted and repeatable camera calibration," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, November 2013.

[16] C. Becker, R. Howard, and J. Rakoczy, "Smartphone video guidance sensor for small satellites," in *Proceedings of the AIAA/USU Conference on Small Satellites*, pp. 1–8, Aug 2013.

[17] R. C. Hoover, *POSE ESTIMATION OF SPHERICALLY CORRELATED IMAGES USING EIGENSPACE DECOMPOSITION IN CONJUNCTION WITH SPECTRAL THEORY*. PhD thesis, Colorado State University, 2009.

[18] H. Fekrmandi, S. Rutan-bedard, A. Frye, M. Yoon, and R. Hoover, "Vision-based guidance and navigation for swarm of small satellites in a formation flying mission," 2019.

[19] H. Fekrmandi, S. Rutan-Bedard, A. Frye, and R. Hoover, "Validation of vision-based state estimation for localization of agents and swarm formation," in *USCToMM Symposium on Mechanical Systems and Robotics*, pp. 216–224, Springer, 2020.

[20] F. Hadaegh, B. Acikmese, D. Bayard, G. Singh, M. Mandic, S. Chung, D. Morgan, K. Alfriend, M. Akella, J. Hurtado, *et al.*, "Guidance and control of formation flying spacecraft: From two to thousands," *Adventures on the Interface of Mechanics and Control*, pp. 327–371, 2012.

Review & Comments for CubeSats linear dynamics paper

Sevil, Hakki Erhan
University of West Florida
hsevil@uwf.edu

Estimated Utility of:
"Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation"
Utility to Community: $5
Utility to Society: $5

The paper presents linear dynamics and controller design for small spacecraft - CubeSats, and orbital position results are given in simulation environment. The following comments and concerns help authors revise their paper and should be addressed before final submission.

The title of the paper needs a revision. There is no state estimation presented in the paper.

Page numbers should be added to paper.

All section titles are all-caps, except "Introduction" section. That should be all-caps as well in order to have consistency.

Page 2, left column, 1st paragraph, line 15: "...(Unmanned Ground Vehicles - UGVs)..." instead of "...(UGVs)..."

Page 2, left column, 1st paragraph, line 21: "...to GPS In the" there is dot missing.

Page 2, left column, 1st paragraph, line 25: "...agents to maintain..." instead of "...agents for maintain..."

Page 2, left column, Section 2, 1st paragraph, line 5: "...shouldn't affect..." instead of "...shouldn't effect..."

Page 2, right column, 3rd paragraph, line 3: "...presented in [19],..." instead of "...created by [19],..."

Page 2, right column, 4th paragraph, line 4: "...[x, y] The state..." there is a dot missing.

Page 2, right column, 5th paragraph, line 5: "...for the purpose of..." instead of "...for the purposes of..."

Page 2, right column, 6th paragraph, line 4: "...holds as long as..." instead of "...holds so long as..."

Figure 2: Third legend has no label.

Page 3, left column, 1st paragraph, line 1: "...a vector of x_0 with size n..." instead of "...a vector of n x_0 blocks"

Page 3, left column, 3rd paragraph, line 10: comma after "satellite 1"

It would be great if there is an explanation of 5 points in the plots. In absolute position plots, they seem to be at the same locations, however in different relative position plots, they are not at same locations. If they don't represent same locations, maybe other symbols would be better to show them in different plots.

Review for: Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation

Filipovich, Roman
Apple
romanmail@apple.com

Estimated Utility of:
"Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation"
Utility to Community: $30
Utility to Society: $50

The paper describes a controller for formation flying of satellites and an underlying dynamics model. This is an interesting paper with a good introduction to the field.

Several things the authors may wish to consider in the next iterations of this work:

- it would be good to understand better how the proposed dynamics model is related other linear models proposed in the past, if any.
- I am not familiar with the field and am curious if generalization of the model beyond the experimental setup described in the paper is something to consider.

Very relevant topic, but focus of work matches neither paper title nor primary challenge identified in introduction

Wilde, Markus
Florida Institute of Technology
mwilde@fit.edu

Estimated Utility of:
"Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation"
Utility to Community: $5
Utility to Society: $5

In summary, the paper argues that linearized models and controllers are needed to enable formation flight control in CubeSat swarms. It presents an unconventional linearization approach for this problem. However, it does not compare this approach against other established approaches such as the Clohessy Wiltshire equations and does not provide validation against the actual two-body equations. It is also not addressed how this model can contribute to the area of distributed state estimation in CubeSat swarms, although it hints at maybe using visual sensing methods.

The following elements of the paper should be considered more:

1. Most of the missions in Table 1 that are listed as "ongoing and recently completed" are actually just concepts or planned missions
2. On page 2, it is stated that the nonlinear model is a single-body model. The proper term is two-body model: Body 1 is the Earth, body 2 the spacecraft
3. On page 2: "These limitations shouldn't effect..." must be "...affect..."
4. On page 2, it is stated that a satellite in a 6,317 km orbit is 100 km above the Earth's surface. However, the mean radius of Earth is 6378 km. In addition, at 100 km altitude, the assumption of negligible drag is no longer valid, as the drag will be large enough to basically make the satellite fall like a rock.
5. The authors state that the linear model "perfectly" recreates the nonlinear dynamics. I expect to see some objective measurements of performance, such as errors in position and velocity.
6. I've never before seen the orbit equations for individual satellites in a swarm be linearized before analyzing formation flight. This approach is completely uncommon in the field of spacecraft dynamics. Commonly, the relative equations of motion are linearized using the Clohessy Wiltshire equations. How doe the results of your linearization approach and the results of your controller study compare against using the CW equations?
7. Again, running the simulation for a 100 km orbit makes no practical sense
8. What maximum levels of thrust are assumed in your controller? Are those realistic for CubeSats?
9. The relative trajectories shown in Figure 4 do not match those I'd expect from the CW equations. Please explain.
10. What is the difference between Figure 3 and Figure 6?
11. The title of your paper is "Validation of distributed state estimation..." Where does your research actually perform state estimation? From what I can see, you assume full state knowledge in a feedback control loop. This does not at all tackle the problem of relative state estimation in a satellite swarm, which is a very pertinent and important challenge.
12. In both the introduction and the conclusion, you mention inter-connectivity and state estimation in a GPS denied environment. However, the body of the paper does not address this at all. I recommend reworking intro and conclusion to more properly reflect the focus of your contribution.

Revisiting the Pull Test for the Measurement of Static Coefficient of Friction

Oren Masory
Department of Ocean and Mechanical Engineering
Florida Atlantic University, Boca Raton, FL, USA
masoryo@fau.edu

Abstract
Measuring static coefficient of friction (SCOF) by pull test, in particular in wet conditions, is considered by many unreliable. This paper describes the results of 2880 pull tests that were performed by 12 teams of inexperienced individuals. Each team tested 2 tiles under three conditions: Dry, Wet (tap water) and Soaped water. For each case 10 pulls in the 4 cardinal directions (total of 40) were taken. The results indicate that if the tests are executed correctly (according to ASTM C-1028) the SCOF in the wet conditions are well distinguished from the results obtained in dry conditions. And as expected, the results for the SCOF for the soaped condition were even lower than the one for the wet conditions.

I. INTRODUCTION

The first Pull Test, for the determination of SCOF, was performed by Leonardo da Vinci at the end of the 15th century [1, 2] and over the years it was formulized by Amontons, Coulomb and others [3]. Pull test were used for many years to determine the SCOF of walking surfaces and several standards, describing the procedure by which the test has to be executed, were established [e.g. 4-6].

This test method was criticized, in particular, on two issues [7]:

1. Repeatability – Since the test is performed manually, the concern is that an operator might introduce error due his inability to repeat the test. Or, in case of more than one operator the results will not be same due to the way the individual performs the test.
2. Test of wet surfaces – While the test's procedure is acceptable for dry surfaces, the concern is that when wet surface is tested the results are affected by a phenomena called *sticktion*, which results in higher values of SCOF for wet surface than of dry surface.

Sticktion, as applied to wet testing of walking surfaces, arises as a function of the test slider's contact time on the surface. If there is any delay between the instant of slip meter pad contacts the surface and the application of the horizontal force (the pulling force) *sticktion* will occur. When a shoe hits a walking surface the interaction time is very short. On wet surfaces, when the contact time shorter than 0.2 seconds (200 milliseconds) significant *sticktion* is produce that can result in slip meter readings that are higher under wet conditions than would be obtained on the same surface in a dry state.

Some commercial slip meters such as English XL, BOT-3000, British Pendulum Tester (BPT), Brungraber Mark II and MARK IIIB and others claim to resolve the above issues. These devices require minimum training and the value of the measured SCOF is immediately and explicitly provided to the user. Publication that compared the performance of these devices reported large discrepancies between the SCOF values obtained by the different devices [8-20].

This paper revisits the commonly used pull test and provides the results of a large number of pull tests which give some insight to the issues above.

II. TEST PROCEDURE

48 inexperienced individuals, with little background in engineering, were randomly divided to 12 team of 4 members. Each team tested 2 tiles (see Table 1). Each team was provided with a 3" square 3/8" thick Neolite pad (ASTM D2000 rating AA and 70A shore hardness), a 20[lb] weight and a force gauge (the gauge did not hold the peak reading), bottle of tap water and a bottle of tap water with 0.0625% of commercial detergent.

Six different tiles were used in these experiments (see Figure 1):

1. Ceramics
2. Vinyl
3. Porcelain with fine sandy texture
4. Glazed ceramics with grooves
5. Polished marble
6. Porcelain with smooth surface.

Table 1: Tiles assigned to each team.

Team	Tiles	Team	Tiles
1R	1,5	4R	2,4
1T	3,5	4T	2,6
2R	2,4	5R	5,6
2T	1,2	5T	1,5
3R	3,6	6R	3,4
3T	1,6	6T	3,4

The instructions provided to the teams were simple:

a. Calibrate the force gage using accurate weights.
b. Measure the weight, used as a load, using an accurate electronic scale.
c. Place the pad on the tile and mount the weight on top the pad.
d. Pull the assembly parallel to the tile's surface and measure the force needed to initiate a motion.
e. For the wet and "soaped" surfaces make sure to flood interface between the pad and the tile and perform the test as soon as possible after the flooding.
f. Repeat the test 10 times in four perpendicular direction of the tile (total 40 reading for each tile)
g. Perform (e) for the conditions: Dry (D), Wet with water (W) and with soaped water (S).

Thus, 160 pull tests were performed on each tile for each of the conditions Dry (D, Wet with water (W) and wet with soaped water (S). In total 2,880 pull tests were carried out in this experiment.

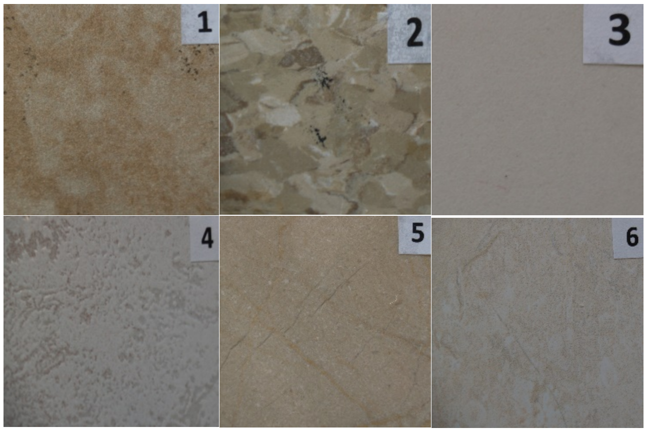


Figure 1: Photos of the 6 tiles.

III. COF Tests' Results

The results, using all data are shown in Figure 1. It shows the average value of the SCOF obtained for each tile (160 readings for each team).

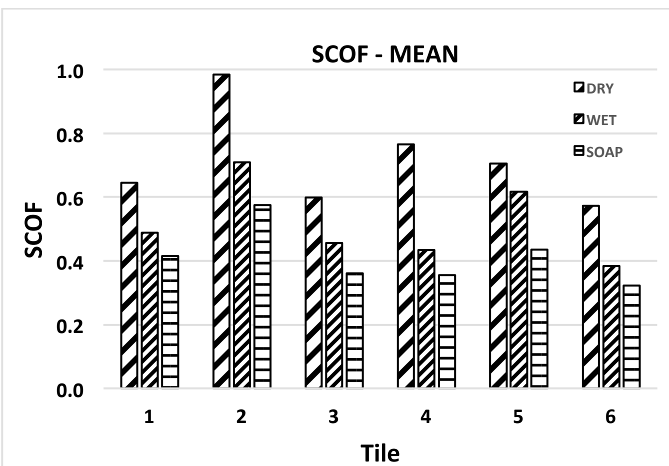


Figure 2: SCOF's mean for the six tiles

The results obtained for all tiles in the respected conditions are the expected ones. Meaning that the SCOF on water wetted surface is lower than in dry conditions and higher than the one with soap.

However, these results contradicts the commonly accepted concern that due to *stickion* the reading of the SCOF in wet condition using pull test will be higher than the reading in dry conditions.

The standard error for each of the cases was calculated and is shown in Table 2. As shown the standard error is very small compared to the mean value of the SCOF. Also, for all tests, the largest values of the standard error are in the pull tests under dry conditions. This due to the fact that the peak value of the pulling force is not recorded automatically by the force gauge and has to be "captured" visually.

Table 2: Standard error for the SCOF of all tiles and test conditions,

Tile	DRY	WET	SOAP	ALL
1	0.0148	0.0098	0.0102	0.0116
2	0.0211	0.0114	0.0113	0.0146
3	0.0125	0.0069	0.0069	0.0087
4	0.0169	0.0105	0.0088	0.0120
5	0.0121	0.0099	0.0072	0.0097
6	0.0114	0.0113	0.0061	0.00961

Figures 3 show results obtained for each tile and by each team and Table 3 provides the corresponding standard error for each case.

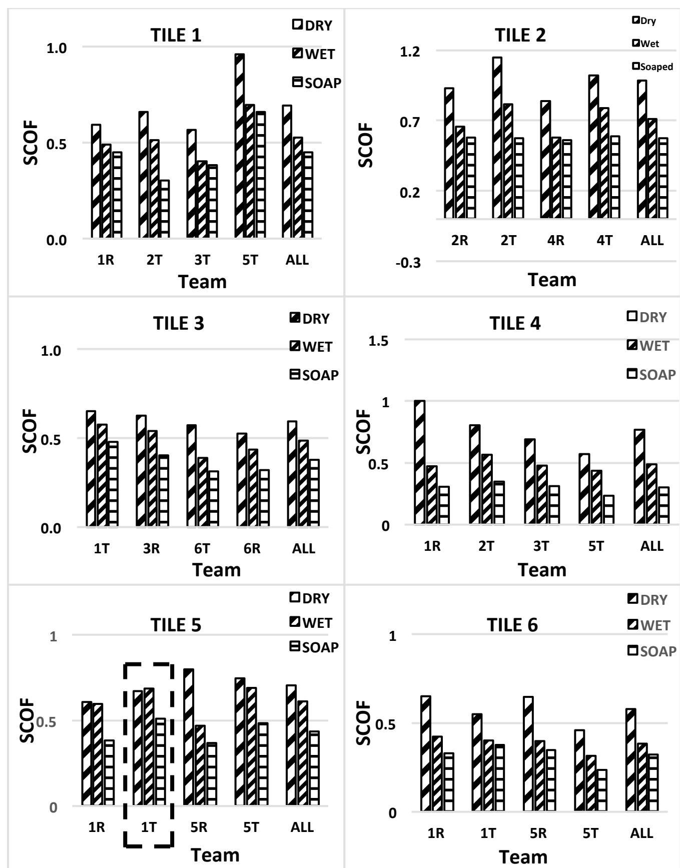


Figure 3: Tests' results for all tiles.

From Figure 3 and the values in Table 3, it could be concluded:

a. As expected the SCOF of the dry tiles is higher than the one of the wet ones, except of one case – Tile 2 measured by group 1T.
b. As expected the SCOF of the water wetted tiles is higher than the one wetted with soaped water.
c. In some cases there are large differences between the SCOF's values obtained by different teams. For example Tile 1 Team 5T obtained SCOF=0.96 while Team 3T obtained the value of 0.57.
d. The standard error for all cases, shown in Table 3, is one order of magnitude lower than the nominal value of the SCOF.

Table 3: Standard error for each test shown in Figures 3.

TILE	TEAM	STANDARD ERROR			TILE	TEAM	STANDARD ERROR		
		DRY	WET	SOAP			DRY	WET	SOAP
1	1R	0.022	0.015	0.027	4	2R	0.022	0.010	0.017
	2T	0.024	0.023	0.013		4R	0.036	0.016	0.017
	3T	0.022	0.025	0.016		6R	0.023	0.023	0.017
	5T	0.041	0.012	0.020		6T	0.053	0.034	0.017
2	2R	0.040	0.013	0.013	5	1R	0.020	0.018	0.016
	2T	0.036	0.018	0.019		1T	0.015	0.013	0.016
	4R	0.037	0.019	0.025		5R	0.032	0.018	0.016
	4T	0.055	0.040	0.033		5T	0.030	0.029	0.016
3	1T	0.016	0.008	0.018	6	3R	0.018	0.016	0.015
	3R	0.018	0.016	0.010		3T	0.026	0.017	0.015
	6R	0.016	0.011	0.016		4T	0.031	0.018	0.015
	6T	0.053	0.014	0.019		5R	0.016	0.022	0.015

IV. Effect of Pulling Direction

The manufacturing process of some tiles is directional, e.g. extruded ceramics tiles, while in others, such as polished marble tile, it is not. To find whether or not the SCOF value is influence by the manufacturing process, t-Test was performed for all pair of pulling directions combinations: NE, NS, NW, ES, EW and SW. The test answers the question if the means of SCOF of the pair are the same or not ($P<0.05$). Table 4 summarized the results obtained by the t-Test: S means that statistically the pair have the same mean and D means that statistically that the means are different.

Although the results of the t-Test indicate that the pulling direction does affect the value of the SCOF in some cases, at least from statistical point of view, one has to realize that practically the differences are small and irrelevant when the SCOF is measured on actual sites where the conditions to perform the test are not ideal. Figure 4 shows the mean values of the SCOF for the four perpendicular direction of the pulls. As can be seen the differences are very small and considering the errors in the testing procedure the errors are not significant.

V. Teams' Performance

The performance of each team can be measured by the standard errors of their results. Thus, the results given in Table 2 were rearranged and plotted in Figure 10. A small value for the standard error indicates more attention and strict following

Table 4: t-Test results.

TILE	CONDITION	DIRECTIONS					
		NE	NS	NE	ES	EW	WS
1	DRY	S	S	S	S	S	S
	WET	S	S	S	S	D	D
	SOAP	S	S	S	S	S	S
2	DRY	S	S	S	S	S	S
	WET	S	S	D	S	D	D
	SOAP	S	D	D	S	S	S
3	DRY	S	D	S	S	S	D
	WET	S	S	S	S	S	S
	SOAP	S	D	D	D	D	S
4	DRY	S	S	S	S	S	S
	WET	S	D	D	S	S	S
	SOAP	S	S	S	S	S	S
5	DRY	S	D	S	D	S	S
	WET	D	S	S	S	S	S
	SOAP	S	S	S	S	S	S
6	DRY	D	S	S	D	D	S
	WET	S	D	D	D	D	S
	SOAP	D	S	S	S	S	S

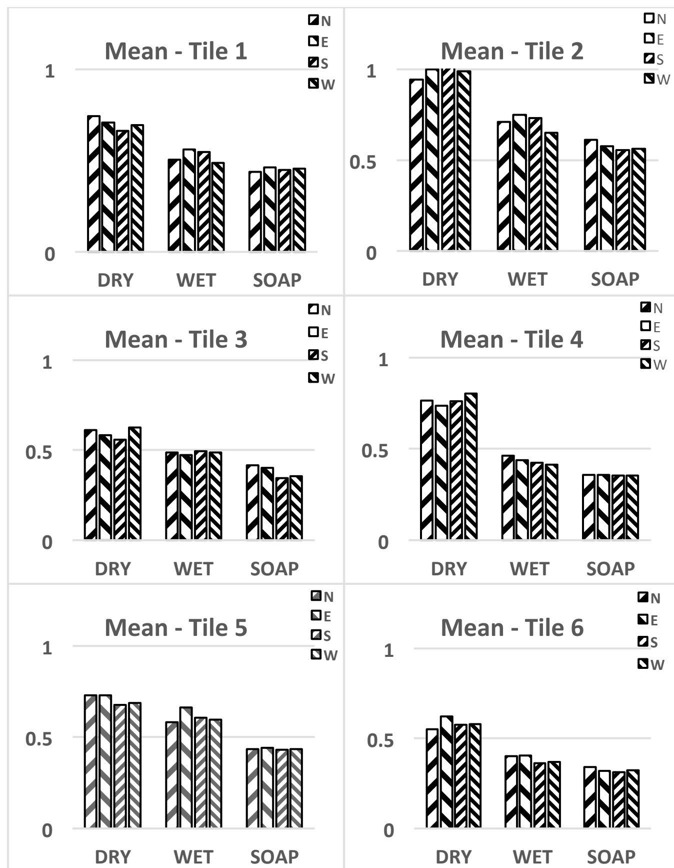
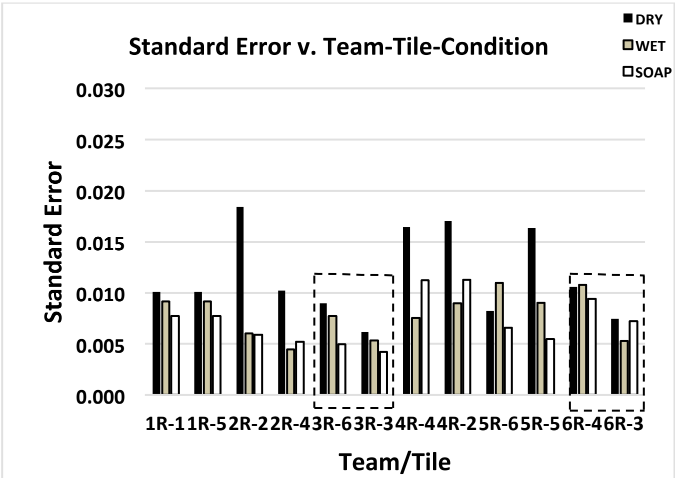


Figure 4: SCOF for all tiles at different pulling directions.

of instructions (see framed cases in Figure 10 teams 3R, 6R and 1T). On the other hand a large value of the standard error indicates poor attention to the testing procedure (see Figure 10 teams 4T and 6T).



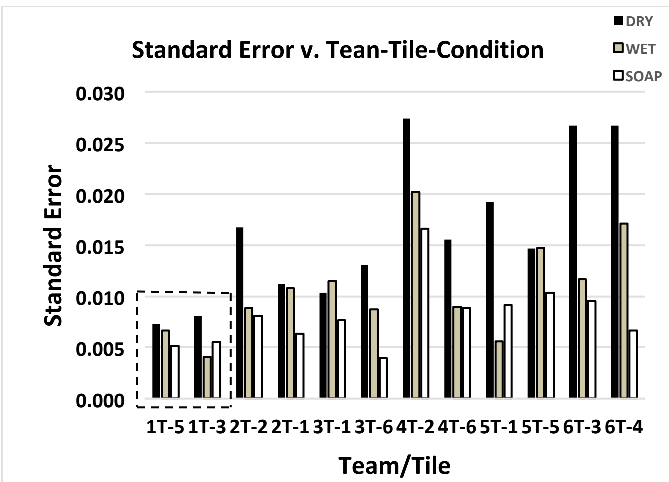


Figure 10: Standard error by team and tiles.

VI. CONCLUSIONS

The results of a large number of pull tests that were performed by inexperienced operators on 6 different tile in three different surface conditions have been presented. The results lead to some conclusions:

a. The pull test is valid for wet surfaces given it is performed correctly.
b. Training will improve the results.
c. The use of the right force gauge, which records the peak value of the pulling force, will improve the results.
d. The number of tests cannot be fixed and should increase to the point where the standard error reached a certain value (e.g. 5%).
e. The test does not require expensive equipment.

VII. REFERENCES

[1] Ian M. Hutchings, "Leonardo da Vinci's Studies of Friction", Wear, 360-361 , 2016, pp. 51-66. http://orthoinfo.aaos.org/topic.cfm?topic=A00130

[2] Tribology History, http://www.tribonet.org/tribology-history/.

[3] Elena Popova, Valentin L. Popov, " The research works of Coulomb and Amontons and generalized laws of friction", Friction, 3(2), 2015, pp. 183-190.

[4] ASTM C 1028-96," Standard Test Method for Determination the Static Coefficient of Friction of Ceramic Tile and Other Like Surfaces by Horizontal Dynamometer Pull-Meter Method.

[5] ANSI/NFSI B101.1 2009, "Test Method for Measuring Wet SCOF of Common Hard-Surface Floor Materials.

[6] ASTM F 609 -96, "Standard Test Method for Using a Horizontal Pull Slip meter (HPS).

[7] "Inherent weaknesses in ASTM C1028 – 07 SCOF Test Method", https://www.c1028.info/weaknesses.

[8] Jungsoo Kim, "Comparison of Three Different Slip Meters under Various Contaminated Conditions", Safety and Health at Work, Vol. 3, No. 1, March 30, 2012.

[9] Kai Way Li, et. al, "Evaluation of two models a slipmeter", Safety Science, Vol. 47, 2009.

[10] Dennis Chimich, et. al.,"Variability of Walkway Tribometer Measurements", https://app.oxfordabstracts.com/stages/123/programme-builder/submission/22562?backHref=/events/123/sessions/52

[11] Wen-Ruey Chang, Simon Matz, "The slip resistance of common footware materials measured with two slipmeters", Applies Ergonomics, Vol. 32, 2001.

[12] A comparison of two slipmeters, ZURICH,Risktopic 4-3.010 October 2012.

[13] Raoul Gronqvist, et. al," Evaluation of three portable floor slipperiness testers", Industrial Ergonomics, Vol. 25, 1999.

[14] Kevin Hallas, Robert Shaw," Evaluation of Kirchberg Rolling Slider and SlipAlert Slip resistance meters", Healt & Safety Laboratories, Harpur Hill, Buxton, Derbyshire, SK17 9JN, 2006

[15] Wen-Ruey, et. el, "Field evaluation of two commonly used slipmeters", Applied Ergonomics, Vol. 34, 2003.

[16] C. M. Powers, et. al,, "Repeatability and Bias of Two Walkway Safety tribometer", Journal of testing and evaluation, ASTM 1999.

[17] Brian C. Grieser, et. el, "Slip Resistance - field measurements using two modern slipmeters", Professional Safety, June 2002.

[18] Raoul Gronqvist, et. al, "The validity and reliability of a portable slip meter for determining floor slipperness during simulated heel strike", Accident Analysis and Prevention, Vol. 35, 2003.

[19] C. M. Powers, et. al,, "Validation of Walkway Tribometers-Establishing a Reference Standard by Powers", Forensic Sciences, Vol. 55, 2010.

[20] Oren Masory, "Comments On ASTM-F2508-13", Journal of Ergonomics, September 30, 2016.

Review & Comments for SCOF pull test paper

Sevil, Hakki Erhan
University of West Florida
hsevil@uwf.edu

Estimated Utility of:
"Revisiting the Pull Test for the Measurement of Static Coefficient of Friction"
Utility to Community: $5
Utility to Society: $5

The paper presents static coefficient of friction pull test, with 12 team and 2880 total tests. The tests include different tiles, different surface conditions, and different pull directions. The following comments and concerns help authors revise their paper and should be addressed before final submission.

The main concern is the relevancy of the paper with the conference areas. The authors need to clarify which area listed under the CFP their paper fits in

There is an empty page at the end of the paper (Pg. 5)

On Pg.1, left column, section 1, 3rd paragraph, line 2 and 3: there are two "ands", one should be removed

Table 1: There is no description about what "R" and "T" stand for in team naming.

On Pg.2, left column, section 3, 1st paragraph, line 1: "in Figure 2." instead of "in Figure 1."

Table 2: "Average" would be better for last column label instead of "ALL"

Figure 3: The x-axis labels of sub-figures for Tile 4 and 6 should be corrected (they are same as Tile 1).

Table 3 caption: "in Figure 3" instead of "in Figures 3".

On Pg.3, left column, section 4, 1st paragraph, line 6: one of the directions is given as "SW", but in Table 4, it's given as "WS". They should be same for consistency.

Table 4: Third direction label is given as "NE", that should be "NW".

Figure 10: The title of 2nd sub-figure: "Team-Tile-Condition" instead of "Tean-Tile-Condition".

Conclusion: There is an empty list entry "e".

Review

Morris, Melissa
Embry-Riddle Aeronautical University
morrim44@erau.edu

Estimated Utility of:
"Revisiting the Pull Test for the Measurement of Static Coefficient of Friction"
Utility to Community: $20
Utility to Society: $60

In summary, the paper argues that pull tests can be performed by inexperienced users, with caveats. It's claimed contributions are:

1. Adding to knowledge of pull test use and accuracy
2. Inexperienced users can get usable results from a pull test

There are some minor typos - a missing) on page 2, the conclusion contains an "e" point that isn't listed, and a few even minor things elsewhere.

Good experiment, but the application of the data is not clear.

Wilde, Markus
Florida Institute of Technology
mwilde@fit.edu

Estimated Utility of:
"Revisiting the Pull Test for the Measurement of Static Coefficient of Friction"
Utility to Community: \$5
Utility to Society: \$5

The paper presents a comparison of pull test results with a large data set. The data is presented well, but no clear conclusion as to the different test methods and potential areas of improvement is provided. In general, I do not fully understand what the contribution is. Was this test series supposed to just collect more data for other researchers? Can the results be applied by manufacturers of test equipment? And very importantly, what is the connection to robotics and control systems, which is the focus of the conference?

The following elements of the paper should be considered more:

1. You use the term "sticktion", but it should be "stiction"
2. A very large number of typos and grammatical errors that distract from the material presented.
3. It would be good to show the equation used to convert the force measurement into the SCOF.
4. Is the "standard error" reported in Table 2 the standard deviation in the data?
5. This is partially addressed in section V, but there is more to be done: How big is the standard deviation for measurements taken by the same team? Is the difference between the teams bigger than within teams? Do you see any trends for the individual teams? Maybe one team consistently produces higher SCOF values than the others. Is there a reason why? Are there any quirks in the technique that can produce higher or lower results?
6. What is the actual contribution of this research? How do you results compare against the existing studies? How can I apply your data to make the SCOF tests better? How does the quality of your results compare with commercial slip meters?

Wave propagation in 3D printed parts with different internal characteristics

Alireza Modir, Ibrahim Tansel

Mechanical and Materials Engineering Department, Florida International University

Abstract

Structural health monitoring in additively manufactured parts is gaining so much attention due to the vast application of 3D printing parts in different industries. In this paper surface response to excitation (SuRE) method as a low-cost SHM technique is used for load monitoring in structures fabricated by Additive Manufacturing (AM). For this purpose, Fused Deposition Modeling (FDM) is implemented for fabricating a round part made of Polylactic Acid (PLA). The infill ratio is the parameter helping for the detection of the applied load on the structure. The part has four sections with different internal design (20, 40, 60 and 100% infill densities). One PZT is attached in the middle of the part for exciting the structure with a sweep sine in a wide range of frequencies (100-300 kHz). The response of the structure in each part is monitored simultaneously by attaching one PZT at the end of the centerline in each section. Results reveal that SuRE method can effectively detect the loading in additively manufactured parts. The monitored response of sensors at sections with different infill ratio helps to detect the loading location.

Keywords: Structural Health Monitoring, SHM, Damage Detection, Additive Manufacturing

1. Introduction

Structural health monitoring(SHM) methods are used for evaluating the condition of a structure in terms of defects such as the crack growth, delamination and external force. In recent years, SHM has gained a lot of attention from researchers in different fields including aerospace, civil, marine, and military [1]. Employing SHM has various beneficial aspects such as improvement of public health and safety, enhancing the life span of the structures and reducing the maintenance cost [2, 3]. Structural Health Monitoring approaches can be divided into two main groups; active and passive SHM. In passive SHM, different operational parameters are measured and the condition of the structure is concluded based on these parameters. Although passive SHM is helpful, it does not reveal the condition of the damage directly. On the other hand, active SHM helps to have online monitoring of the structural health and defect detection [2]. The biological nervous system was an inspiration for researchers to work on SHM [4]. Each SHM system normally contains three main parts: sensors, data acquisition system, and health evaluation part [5]. Rafieipour et al. [6] Proposed a distributed frequent itemset mining using data in a wireless sensor network. Sensors form a network and some close nodes form a cluster.

Additive manufacturing (AM) also known as 3D printing, has been rapidly developed for manufacturing complex industrial parts that could not be made with conventional manufacturing methods. The most common AM technologies are Fused Deposition Modeling (FDM) technique which also called Fused Filament Fabrication (FFF). FDM is one of the simplest, cheapest and widely used methods in additive manufacturing [7]. FDM 3D printers use filaments in a wide range of materials such as PLA, TPU, ABS, PET, and ASA [8].

Each additive manufactured part has a specific external and internal configuration. When a piezoelectric sensor is attached to the part, the mechanical response of the part while vibration is induced to the part can be coupled to the electrical response of the sensor [9]. After a signature is established for a defect-free part, the signature of the part with a defect can be compared with the baseline to detect the existence of any defect. There have been so limited studies investigating the SHM of additive manufactured structures. For monitoring the health of the 3D printed parts, Baere et al. [10] designed "a network of pressurized capillaries" inside a 3D printed part and the performance of this method, which was named effective Structural Health Monitoring (eSHM), was based on the changes in the fluid pressure inside the capillaries. Any changes in the pressure can be a sign of a defect in the structure. Surface Monitoring by the aid of image-based sensors is one of the common methods of in-situ monitoring for defect detection in additive manufacturing systems [11]. Comprehensive reviews of different approaches in monitoring metal-based additive manufacturing can be found in [12]. Albakri et. al. [13] used "electromechanical impedance measurements" as a non-destructive evaluation technique for the detection of defects in AM parts. Electromechanical Impedance (EMI) method is a non-destructive SHM method that investigates the changes in electrical impedance for detecting defects in the structure [14]. EMI is a costly method that has not been commercialized yet. In the last 2 decades, many researchers have investigated EMI theoretically [15] and experimentally [16].

Piezoelectric wafer active sensors (PWAS) are significantly improving active SHM in evaluating structural health [2]. These sensors can be attached to the surface of the structure or mounted inside the built-up structure. Most of the active SHM methods excite the surface of the plates with high-frequency surface waves and monitor the propagation of the waves. Since the excited frequencies range is high, the environmental vibration will not affect the measurements. The propagation characteristics of the surface waves depend on the condition of the structure and loading. Change of the propagation characteristics indicates the defect. Surface response to the excitation (SuRE) method is a low-cost and more flexible alternative to the EMI approach [17, 18]. Estimation of the location of the defects has always been a challenge. In this method, generally, one piezo excites the surface of the structure and one piezo monitors the response of the structure to the exciting waves.

Most of the time, engineers prepare hollow parts by using the AM methods to reduce manufacturing time and material cost. The infill type (pattern and density) and the perimeter layer thickness are two important factors in defining the characteristics of a hollow section. In this paper, the wave propagation in AM parts with different infills is studied using Surface Response to excitation (SuRE) method. A part with different infill ratios is fabricated for peformin an experiment in order to identify the response of each section to frequency excitation and detection of external loading. The SuRE method has previously shown an efficient and reliable performance for metal [19] and composite plates and beams [20]. Here we employ this method for wave propagation investigation in additive manufactured parts with different infills and top thickness.

2. Method

The SuRE method, as an active SHM method, can be considered as a low-cost alternative for EMI. Instead of one single PZT, SuRE method uses one PZT for exciting the surface of the structure with guided waves, generally a weep sine signal over a specific frequency range, and one or multiple sensors at desired locations for monitoring the dynamic response of the structure to the excitations. By deploying Fast Fourier Transformation (FFT), the frequency spectrum of the response can be obtained.

The frequency response spectrum will stay stable as far as there is no defect or change in loading is applied to the structure. For detection of defects or changes of the loading conditions, the frequency response of the structure in pristine and faulty conditions are used in the sum of squared difference (SSD) equation as shown in Eq. 1.

$$\mathrm{SSD} = \sum_{\mathrm{i}=1}^{\mathrm{m}} |P_{m\times 1} - F_{m\times 1}|^2 \qquad \text{Eq.1}$$

Here, m represents the size of FFT, P is the FFT of the baseline and F is the FFT of damaged or loaded structure. As can be seen in Eq. 1, the difference in FFT of baseline signal and damaged signal are added to quantify the change for a better comparison.

3. Experimental setup

The round disk in Fig. 1 was fabricated additively by using Polylactic Acid (PLA) using a QIDI TECH X-PRO 3D printer. Slic3r Prusa software was employed for internal design and slicing process of the model. Two different round disks were printed with the same geometry (180 mm diameter and 10 mm thickness) with different internal sections. The disk had four internal sections with rectangular infill patterns in a wide range of infill densities (20, 40, 60 and 100)

As can be seen in Fig.1, five piezoelectric disks were bonded permanently to the surface of the part. One PZT at the center of the part as the exciter and one PZT disk near the edge of each of the four sections for monitoring the dynamic response of the structure. One hole is designed at the center of each section to apply a constant compressive force by tightening a bolt. All the PZT disks used in this research, for excitation and sensing, were Steminc (SMD15T07F3000R) with a 15 mm diameter and a 0.7 mm thickness. Excited sweep sine waves at the central PZT were generated by Rigol DG1022 function generator and Owon XDS3104AE digital oscilloscope was used for monitoring the signal of the piezoelectric disks at each section of the part. The FFT of the collected data was calculated by a MATLAB program. More detail of the experimental setup can be found in [21]. The diagram of the experimental setup is shown in Figure 2.

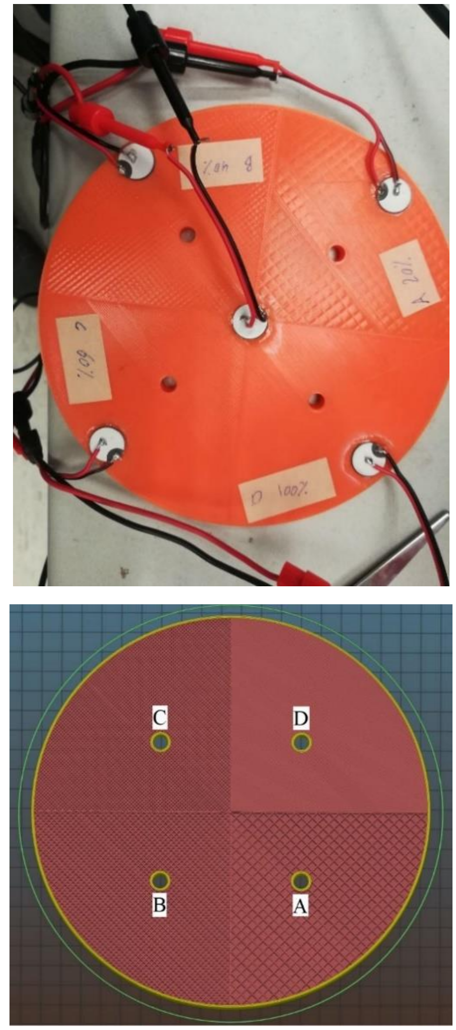


Figure 1: 3D printed round part with four different infills.

The amplitude of excited sweep sine signal was 20 Volts pick to pick with the starting frequency set as 100 kHz and stopping frequency set as 300 kHz. The sweeping time duration considered as 1 millisecond. The experiments conducted in 5 different stages. First, data were collected when the structure was in a relaxed condition (baseline data). Then the response of the structure was recorded after tightening a bolt at a fixed torque (5kN.m) in each of the four sectors of the part. The load monitoring performance of the SuRE method is studied by utilizing bolts in different locations.

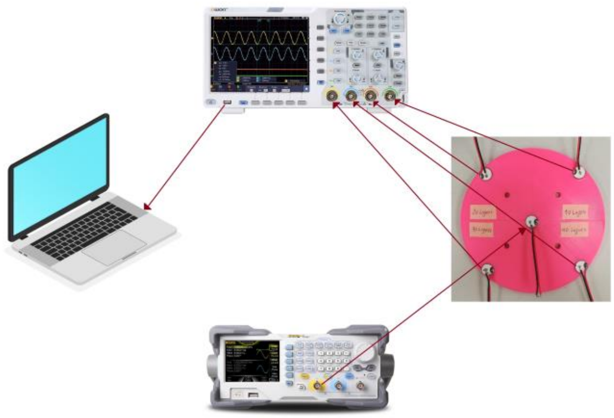


Figure 2: Experimental Setup

Results

The results are presented here for investigating the effect of infill density on wave propagation in 3D printed PLA structures. Evaluating the effectiveness of the SuRE method on load monitoring of additive manufactured parts is also studied. For Implementing SuRE method, first the central piezoelectric was excited with a sweep sine in a broad range of frequencies (100-250 kHz). Then the response of the structure was acquired by four PZTs which were bonded to the surface of the part at different locations, as a reference for SSD calculation.

Figure 3 shows the reference spectrum of the structure and spectrum of the response at different locations when the bolt was tightened at location A (the sector with the filling density of 20%.). As it can be seen in this figure, the most changes in the response amplitude occurred in sector A in the range of 100-120 KHz.

All experiments for the baseline and loaded structure responses were performed with the same configuration. Figures 4-6 compare the FFT of reference data and when the external compression loading is applied at sections B, C, and D with infill densities of 40%, 60%, and, 100%, respectively

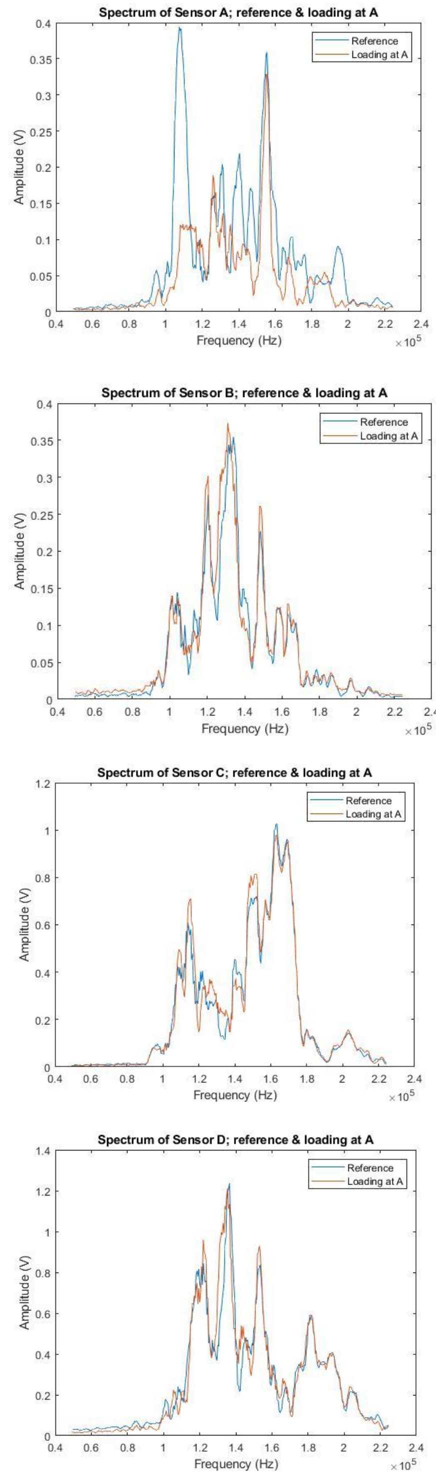


Figure 3: FFT response of the baseline and loaded structure at location A

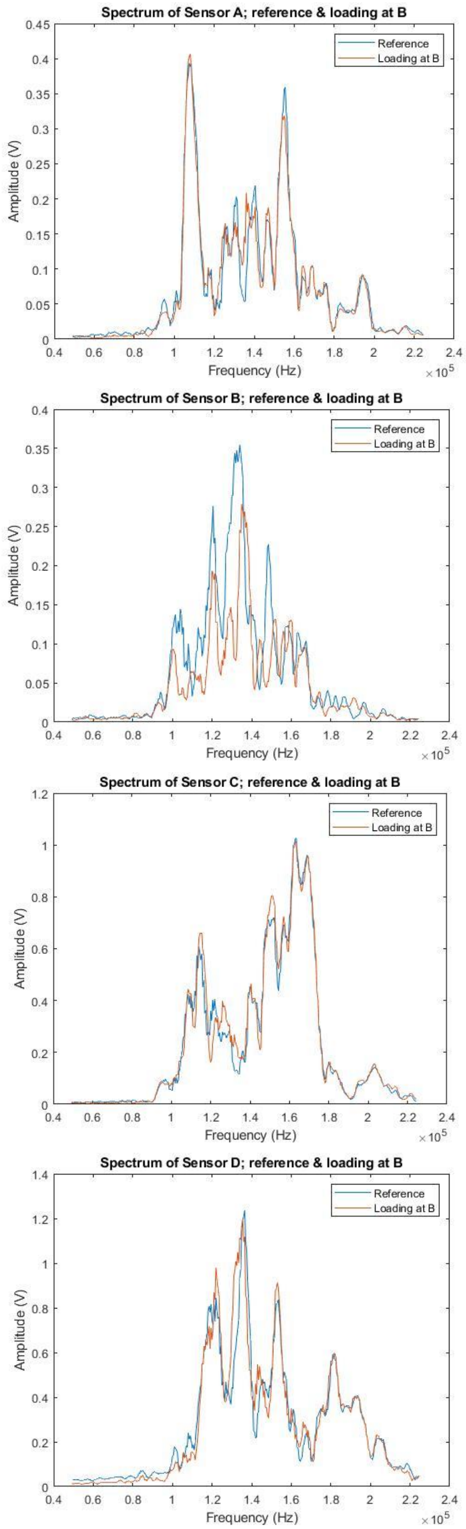


Figure 4: FFT response of the baseline and loaded structure at location B

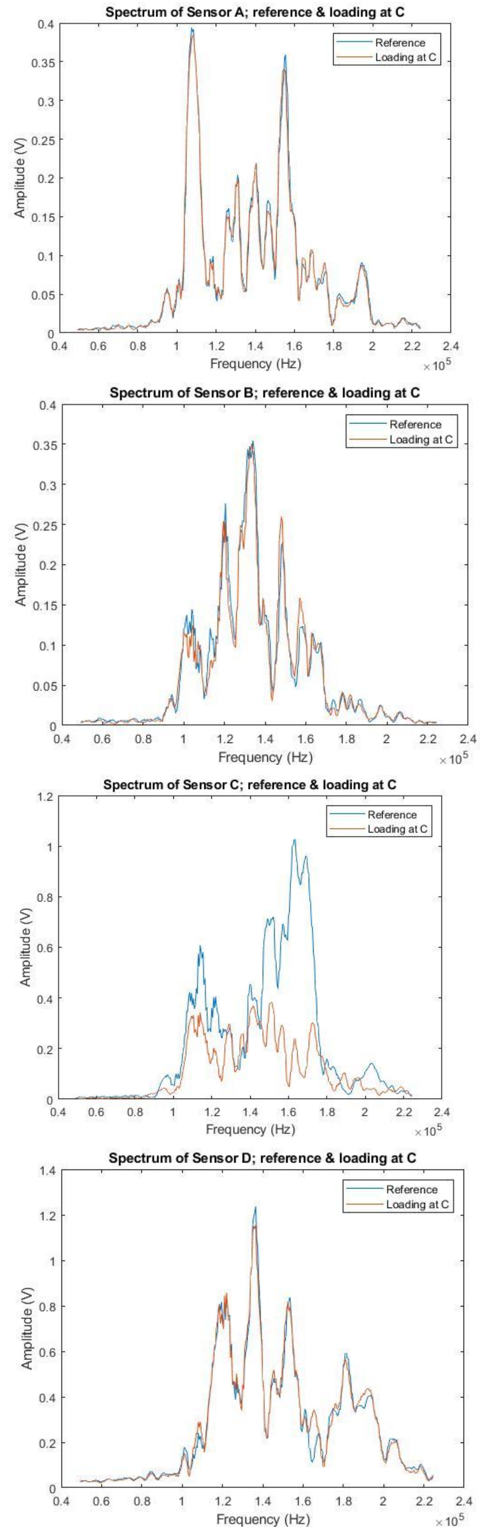


Figure 5: FFT response of the baseline and loaded structure at location C

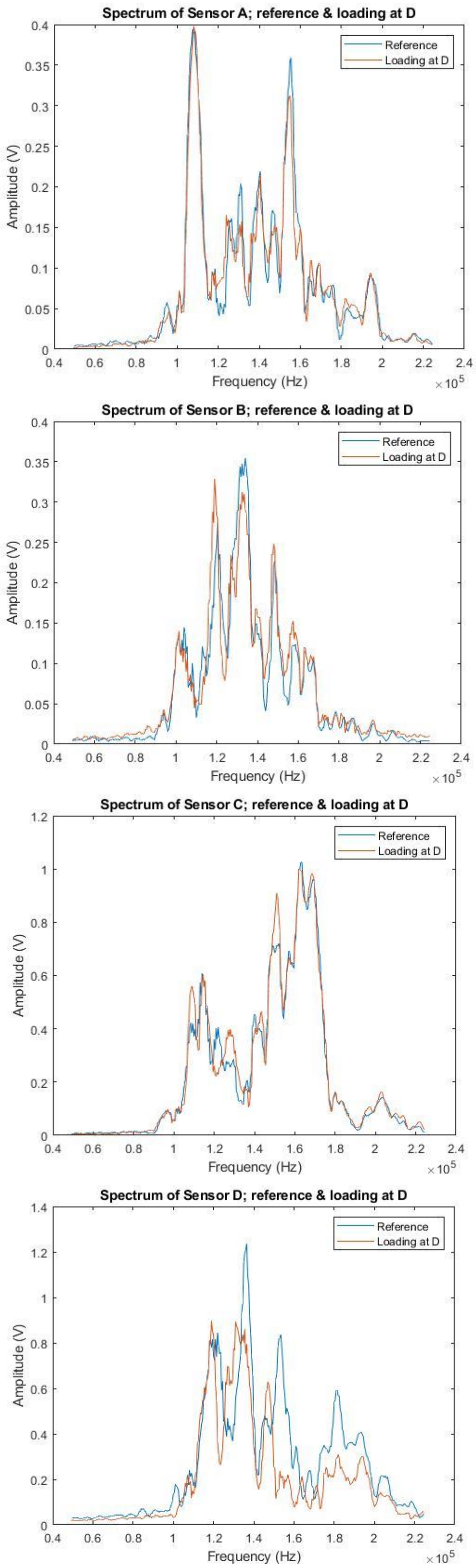


Figure 6: FFT response of the baseline and loaded structure at location D

It can be seen that as the infill increases, the maximum change in the amplitude shifts to higher frequencies. The maximum change in the FFT spectrum for 20%, 40%, and 60% infill densities occur at 100, 130, and 160 kHz, respectively. Fig. 6 shows that when the loading applies on the solid section of the part, with 100% infill density, the changes in amplitude response occurs in a broader range of frequencies.

Table 1 shows the changes in the SSD for different experiments. Generally, by increasing the infill density, the monitored amplitude of the respose increases. In another words, sections with higher ratio of infill, convey the excited waves better. Consequently, higher values of SSD are calculated for sections with denser hidden geometries.

Table 1. SSD for different sensors at different loading locations.

External loading location	SSD for Loading at A	SSD for Loading at B	SSD for Loading at C	SSD for Loading at D
Sensor A	0.091405	0.0066616	0.0012904	0.0077062
Sensor B	0.0098047	0.05696	0.0049782	0.013692
Sensor C	0.0446	0.03558	1.2305	0.04604
Sensor D	0.12844	0.13371	0.027629	0.59884

Conclusion

The purpose of this study was to investigate the performance of SuRE method in load monitoring applications in additive manufactured parts. A round parts made of PLA was fabricated using FDM technology with sections containing infills with different densities. A piezoelectric transducer at the center of each part was employed for exciting the structure with a sweep sine waves in the range of 100-250 kHz.One piezoelectric sensor was attached to the surface of each section for monitoring the dynamic response of the structure to the excited waves in two conditions; when the part is relax and loading is applied at the center of each section. The SuRE method was successful in detecting the load in structures fabricated with additive manufacturing. It was observed that by increasing the infill density, the changes in the response spectrum occurs at higher frequencies.

References

[1] Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., & Cayirci, E, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393-422, 2002.

[2] V. Giurgiutiu, Structural health monitoring: with piezoelectric wafer active sensors, Elsevier, 2007.

[3] Surakanti, S. R., Khoshnevis, S. A., Ahani, H., & Izadi, V, "Efficient Recovery of Structrual Health Monitoring Signal based on Kronecker Compressive Sensing," *International Journal of Applied Engineering Research*, vol. 14, no. 23, pp. 4256-4261, 2019.

[4] N. Sharp, A. Kuntz, C. Brubaker, S. Amos, W. Gao, G. Gupta, A. Mohite, C. Farrar, and D. Mascareñas, "A bio-inspired asynchronous skin system for crack detection applications," *Smart Material and Structure*, vol. 23, no. 5, p. 055020, 2014.

[5] Izadi, V., Shahri, P. K., & Ahani, H., "A compressed-sensing-based compressor for ECG," *Biomedical engineering letters*, pp. 1-9, 2020.

[6] Rafieipour, H., Zadeh, A. A., & Mirzaei, M.,, "Distributed Frequent Itemset Mining with Bitwise Method and Using the Gossip-Based Protocol.," *Journal of Soft Computing and Decision Support Systems*, vol. 7, no. 3, pp. 32-39, 2020.

[7] Wang K, Zhao Y, Chang Y, Qian Z, Zhang C, Wang B, et al. , "Controlling the mechanical behavior of dual-material 3D printed meta-materials for patient-specific tissue-mimicking phantoms," *Materials & Design*, vol. 90, pp. 704-712, 2016.

[8] Mohamed OA, Masood SH, Bhowmik JL. , "Optimization of fused deposition modeling process parameters:a review of current research and future prospects," *Advances in manufacturing* , vol. 3, no. 1, pp. 42-53, 2015.

[9] Sturm, L. D., Albakri, M. I., Tarazaga, P. A., & Williams, C. B. , "In situ monitoring of material jetting additive manufacturing process via impedance based measurements," *Advanced manufacturing*, vol. 28, p. 456463, 2019.

[10] De Baere, D., Strantza, M., Hinderdael, M., Devesse, W., and Guillaume, P., "Effective Structural Health Monitoring with Additive Manufacturing," in *EWSHM - 7th European Workshop on Structural Health Monitoring*, 2014.

[11] Dinwiddie, R. B., Dehoff, R. R., Lloyd, P. D., Lowe, L. E., & Ulrich, J. B., "Thermographic in-situ process monitoring of the electron-beam melting technology used in additive manufacturing.," in *International Society for Optics and Photonics.*, 2013.

[12] G. Tapia, A. Elwany, "A review on process monitoring and control in metal-based additive manufacturing," *Journal of Manufacturing Science and Engineering*, vol. 136, no. 6, 2014.

[13] Albakri, M. I., Sturm, L. D., Williams, C. B., & Tarazaga, P. A. , "Impedance-based non-destructive evaluation of additively manufactured parts," *Rapid Prototyping Journal*, vol. 23, no. 3, pp. 589-601, 2017.

[14] Na, W., & Baek, J., "A review of the piezoelectric electromechanical impedance based structural health monitoring technique for engineering structures.," *Sensors*, vol. 18, no. 5, p. 1307, 2018.

[15] Park, G., Cudney, H. H., & Inman, D. J., "An integrated health monitoring technique using structural impedance sensors.," *Journal of Intelligent Material Systems and Structures*, vol. 11, no. 6, pp. 448-455, 2000.

[16] Annamdas, V. G. M., & Soh, C. K. , "Load monitoring using a calibrated piezo diaphragm based impedance strain sensor and wireless sensor

network in real time.," *Smart Materials and Structures,*, vol. 26, no. 4, p. 045036, 2017.

[17] Mohamed, A. F., Modir, A., Tansel, I. N., & Uragun, B, "Detection of Compressive Forces Applied to Tubes and Estimation of Their Locations with the Surface Response to Excitation (SuRE) Method.," in *2019 9th International Conference on Recent Advances in Space Technologies (RAST) (pp. 83-88). IEEE.*, 2019.

[18] Tashakori, Shervin, Amin Baghalian, Volkan Y. Senyurek, Saman Farhangdoust, Dwayne McDaniel, and Ibrahim N. Tansel, "Composites Bond Inspection Using Heterodyne Effect and SuRE Methods," *Shock and Vibration,* 2019.

[19] Mohamed, A. F., Modir, A., Shah, K. Y., & Tansel, I, "Control of the Building Parameters of Additively Manufactured Polymer Parts for More Effective Implementation of Structural Health Monitoring (SHM) Methods.," in *Structural Health Monitoring 2019*, 2019.

[20] S Tashakori, A Baghalian, M Unal, VY Senyurek, H Fekrmandi, D McDaniel, IN Tansel, "Load monitoring using surface response to excitation method," in *Mechanics of Composite and Multi-functional Materials, Volume 7,* Springer, CHarm, 2017, pp. 209-2014.

[21] Shah, K. Y., Mohamed, A. F., & Tansel, I. N.,, " Additively Manufactured Multi-Material Parts with Defect Detection Capabilities," *Procedia Manufacturing,* vol. 39, pp. 493-501, 2019.

Compliant Joints Actuated Via Cables

Michael Ayer(1), Melissa Morris(2)

(1) Florida Polytechnic University
Department of Mechanical Engineering
Lakeland, FL 33805

(2) Embry-Riddle Aeronautical University
Department of Engineering and Technology
Daytona Beach, FL 32114

morrim44@erau.edu

ABSTRACT
This paper introduces work to investigate the actuation of compliant joints in soft robotics using cable actuators. The general concept is provided. Afterward, the manufacturing of the joints in TPU via 3D printing is investigated with the intent to integrate the cable inside the soft material. The force-deflection relation of a single degree-of-freedom joint is determined for three different raster angles. The results allow for the consideration of printing angle when producing compliant joints via 3D printing and the corresponding optimal actuator selection.

Keywords
Soft robotics, cable actuation

1. INTRODUCTION
Soft robotics has so far primarily involved the use of molded hollow silicone structures actuated by pneumatics. This method is excellent for certain applications and yields a completely soft mechanism. However, there are also problems to be resolved using this material and actuation method including fatigue, risk of puncture, and use in areas with low or no ambient air pressure.

The work presents an alternative actuation method. Cable actuators have been used in robotics for decades [1], [2]. They are a lightweight method of actuation and provide some inherent compliance due to not being completely rigid. As they are a flexible actuation method, they are a suitable method for soft robot actuation [3]. Typically, cable actuation drives rigid links and components, but here the use of soft materials together with compliant links is proposed. An example of how this will work is shown in Figure 1. The soft material can be molded silicone or any other flexible material.

As a first tentative step toward efficiently actuating soft components with cable actuation, this work presents some of the initial investigation into using 3D printed compliant materials. Using this manufacturing technique offers the possibility of faster prototyping and more versatility in design. It also facilitates easy routing of cables by allowing for channels in any shape and location. Work was done here to investigate the force-deflection relationships of TPU that has been 3D printed, with consideration of the raster angle of the print.

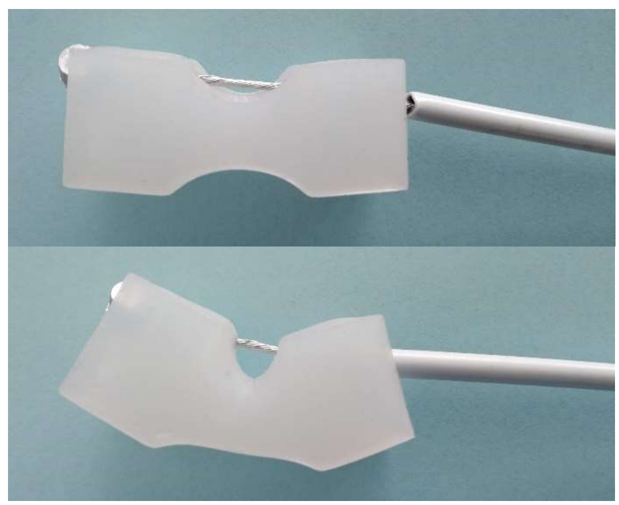


Figure 1. Compliant joint being actuated via cable actuation

This paper begins with some background into compliant links. It then details the goal of the overall research effort. Next, the details of the experiment for the main objective of this part of the work is described. The paper concludes with the result and discussion of the findings here regarding using 3D printed TPU as the compliant component in a cable-actuated soft robot.

2. BACKGROUND
2.1 Literature Review
Compliant mechanisms are an exciting technology that has many advantages. The ability to flex comes from innovative

materials and manufacturing techniques. Compliant materials used as filaments for 3D printers have been addressed over the past few years. This facilitates rapid manufacturing of complex shapes. However, there are still issues that hinder the adoption of printed compliant mechanisms. Traditional mechanisms are designed using materials with known mechanical properties that can be modeled and optimized using Finite Element Analysis (FEA). Unfortunately, this method gives large errors when used design a compliant 3D printed joint. This is because material properties are not well known and studied for this instance. Instead, the design and implementation of 3D printed compliant parts is still mostly done via trial-and-error.

Material considerations are a major issue in compliant joints. Many examples can be found in prosthetic and anthropomorphic hands. Work on an anthropomorphic hand utilizing rubber joints concluded that further research is needed in the material and manufacturing method to increase durability [4]. Another study designed and evaluated a soft prosthetic hand that noted that additive manufacturing methods and technologies have increased the design possibilities [5], but they observed changes in stiffness due to variations in infill density of 3D printed parts. In recent work, an endoskeleton was 3D printed for an anthropomorphic hand [6]. Compliant joints were used with this endoskeleton and different materials were considered with FEA analysis. However, the designed joint was not able to be replicated using 3D printing.

Underactuated mechanisms can combine compliant joints and cables. They have more degrees of freedom than they have actuators, allowing for reduced costs and increased reliability [7]. Used in an anthropomorphic hand, slack cables in an underactuated condition have investigated [8].

Rapid prototyping is being exploited for designing compliant mechanisms. Common inexpensive 3D printers use fused deposition modeling (FDM) [9]. This process leaves behind small gaps, or microchannels, between the material that is extruded. When a compliant part is bent, the gaps become a place for material to go. The orientation of the extrusion, or raster angle, affects the ease of bending at a given angle. Therefore, the mechanical properties of 3D printed parts are very anisotropic and are not well suited for using design methods such as current common FEA abilities [10].

Compliant mechanisms are more difficult to design than rigid mechanisms [11], and as noted above 3D printing can complicate it further. More research is needed to characterize the mechanical properties of compliant materials and how the properties change with varying manufacturing techniques. In addition, standards need to be developed to verify these mechanical properties. In the meantime, designers need to prove the structural and mechanical properties of their designs.

2.2 Overview of Work

The final goal of this project is to eventually develop the means to efficiently design cable-driven soft robotics. There are many factors that will be investigated and incorporated into the models in order to complete the goal. These include the placement and routing of the cables, cable selection, optional sheathing, the shape of the joint, the size of the joint, and the material(s) in the joint. All these things can then be used to help determine the sizing of actuators to meet the goal of optimal actuation. In addition, the manufacturing method needs to be considered, which will affect many of the other aspects.

While the use of mold-formed silicone or rubber is certainly possible and was used here for initial concept testing shown in Figure 1, 3D printing allows for more versatility in the cable channel routing. A channel of any shape can be made in a compliant link, which made all at once with the cable enclosed. The printing process can be halted at key points for the cable to be placed within the joint, perfectly embedding the cable in the desired path within that joint. 3D printing is also becoming more accessible and can be done easily without the need for measuring and mixing raw materials nor vacuum chambers to remove bubbles from molded components.

This paper presents some findings from the first phase of the planned work. Some initial measurements of the material properties of a sample joint were tested. The objective here was to investigate the use of a compliant 3D printed 1-DOF joints with cable actuation. Specifically, the raster orientation effects on the force-deflection relationship was of interest for this part of the work as it is known that this affects the properties of the joint [12].

3. JOINT TESTING

This phase of the project involved investigating the force-deflection relationship of a compliant joint. The raster angle effects in a compliant joint are noted as well. This was done for a simple 1-DOF joint with semi-circular cutouts to create a hinge. The shape was selected for simplicity in order to gain some initial data. A later phase of this work will look at different joint shapes and design and the process used here will be repeated for other shapes.

Due to soft robotics having promise for offering passive safety working alongside humans, there is likely use for robotic manipulators that are human-sized. Therefore, hinge joints with roughly the same dimensions of a human finger were designed for this initial testing project. These joints were printed in TPU with varying raster orientations. The different raster orientations that were tested are: 0°, 45°, and 90° as depicted in Figure 2. The raster angles were produced during the slicing process when the CAD model was being prepared for the 3D printer and are the three most likely to be selected by users who would chose to manufacture a soft robot joint using this method.

For finding the force-deflection relationships, a means was constructed to cause the actuation with a cable. One end of the joint was clamped in a fixed jig while the other end was left free. A cable was then routed over the top of the joint starting from the jig and hanging over the free end of the joint. The cable placement was done in this way as this was meant to be only an initial study and the many possibilities for placement of the cable were not considered during this phase. Calibrated weights were then mounted to the free end of the hanging cable, causing a resultant downward force of the end of the joint. The specimens were all labeled and marked with a center line, which could be visually aligned with a protractor to measure the deflection angle. A close-up of this set-up is shown in Figure 3.

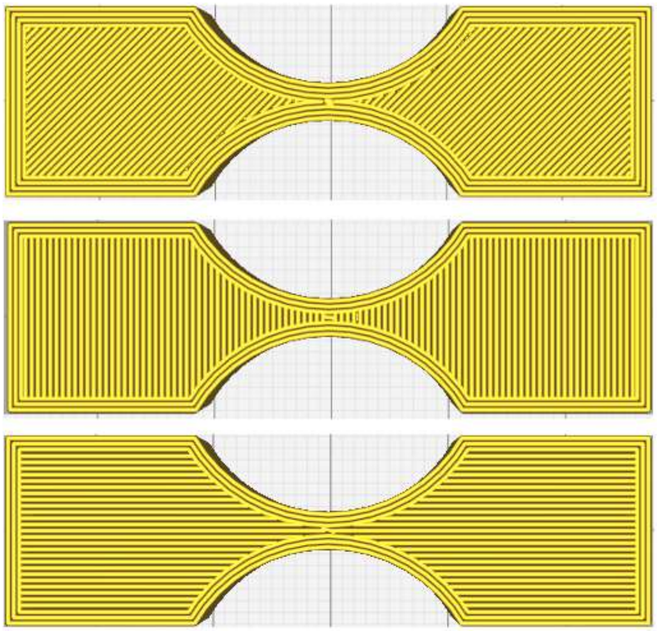


Figure 2. Raster angles at 45° (top), 90° (middle), and 0° (bottom)

The data was captured manually from visual observation of images. These images were captured from a fixed camera mounted a tripod so that the measurements read would be consistent. The camera was calibrated with mirrors before every new session to ensure that the location was exactly the same. An image was captured for every measurement so that it could be carefully checked.

For the testing process, a load was kept on the joint for all measurements. This is because slack cables do not allow for controllable movement. However, the no-load angle of each joint was calibrated to zero because there was some initial bending due it its own weight. This also ensured an even starting point in case there was any irregularities in angle due to how each unique specimen fitted into the clamp of the fixed jig.

4. RESULTS

Based on expectations from literature, there was a difference in the deflections due to the raster angle for compliant 3D printed parts.

An initial test was done at a default infill of around 20% to verify that the raster angle made a difference. The applied force was incrementally increased starting from .4905N to 2.4525N for each of three specimens, after calibration. The maximum force for each was limited by the elastic deformation. Figure 4 summarizes the force vs the deflection for all three raster orientations. There were very clear differences in the curves based on the raster angles as shown.

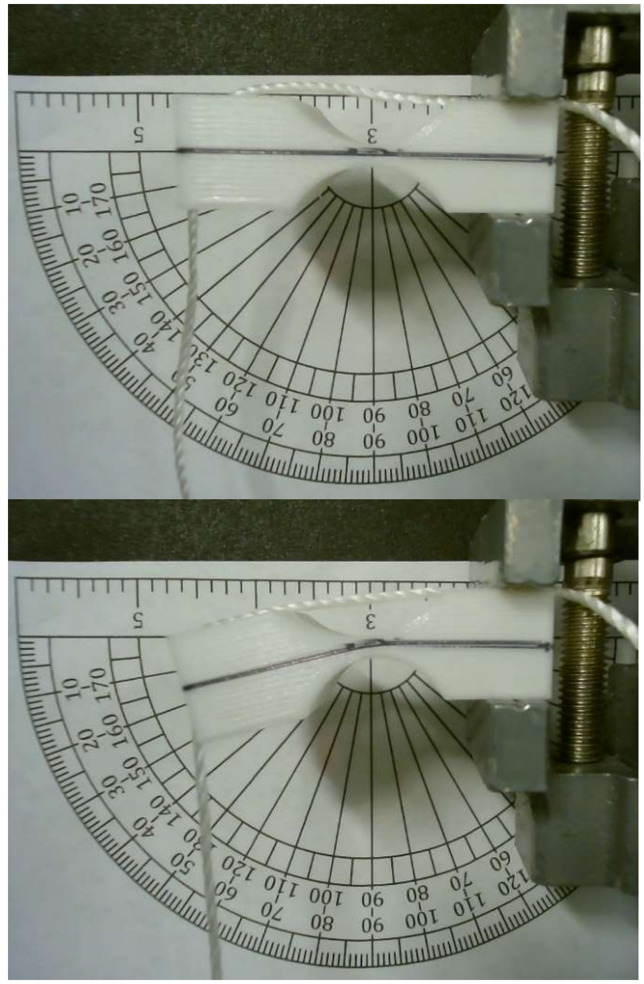


Figure 3. Specimen testing. Unloaded (top) and with a load (bottom)

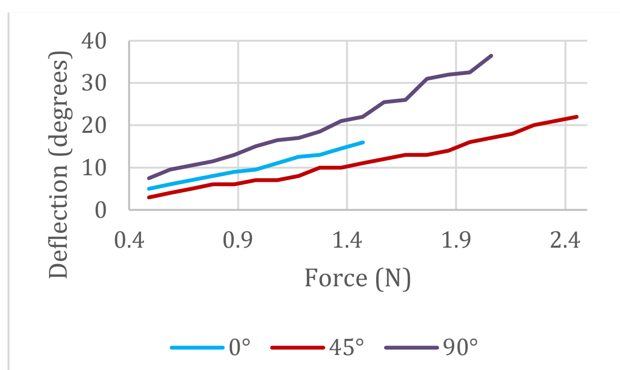


Figure 4. Initial force-deflection test of three joints printed at the raster angles shown via the three colors

The next test involved multiple specimens for each raster angle to further verify the results. In addition to this, another set of tests were done at a different infill in order to gain some feel for how much this affects the results and if there are notable differences in how the raster angles affect the force-deflection compared to the infills. A 20% and 40% infill were chosen due to these both being reasonable settings for sturdy joints without excessive material usage.

Looking at 20% infill test results shown in Figure 5, the general trend follows that of the initial test, but with spread that increases with the applied force. The standard deviation for of samples went from 0.23 at 0.4905N and increased up to 1.79 at about 1.4715N. The increase in deviation was consistent for both the 0° and 90° raster angles. The 45° did not show much variation regarding the deviation between samples, however. These trends are clear from both the figure and raw data.

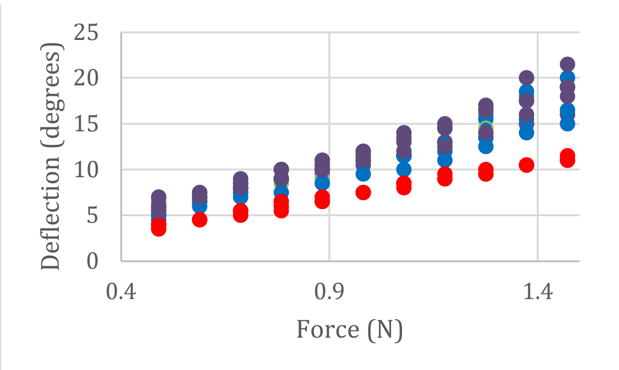


Figure 5. Force-deflection results at 20% infill with raster angles of 0- (blue), 45- (red), and 90-degres (purple)

Figure 6 then shows the tests with a 40% infill. The results here are shown from 0.4905N to 2.1582N. The increased infill allowed for greater force before elastic deformation occurred. As with the lower infill, there is a trend of the samples showing greater deviation and variance as the force increases. With the increased weight, the 45° raster samples showed an increase in deviation as well occurring at 1.4715N. The standard deviations ranged from 0.4 at 0.4905N and increased up to 2.81 at 2.1582N. The rise in the deviation was again not uniform nor perfectly consistent, but the overall trend showed this as can be seen from the figure. Furthermore, the 90° raster showed the greatest spread, as with the lower infill, and it continued to increase faster than the other two angles as the larger forces were applied.

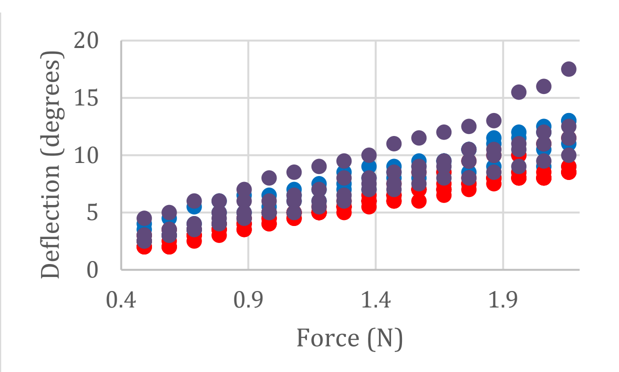


Figure 6. Force-deflection results at 40% infill with raster angles of 0- (blue), 45- (red), and 90-degres (purple)

Finally, a comparison between the infills is made by looking at the averages from each raster angle at each infill. This is shown in Figure 7.

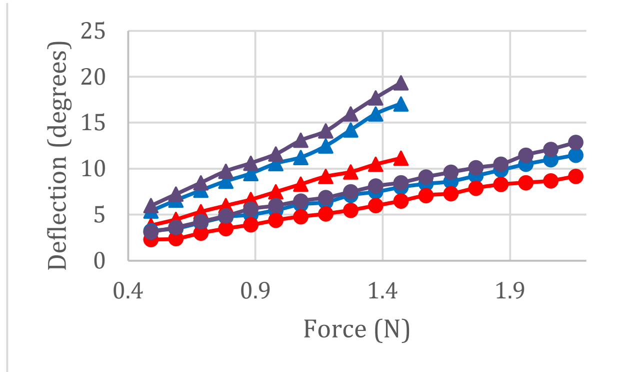


Figure 7. Force-deflection results at 20% infill (triangles) and 40% infill (circles) with raster angles of 0- (blue), 45- (red), and 90-degres (purple)

It is clear from Figure 7 that on average, the increase in infill does cause a difference in deflection. Not only does the joint handle more force with an increased infill, but the deflection angle from the forces are decreased. The raster angles show similar general trends no matter the infill.

5. DISCUSSION

This phase of the experiments showed some general trends that were expected. They are outlined in this section along with some findings that were not necessarily expected.

All tests showed a decreased angular deflection with hinges printed with a 45° raster orientation compared to 0° and especially 90° raster orientations. Increasing the infill did not change this fact, as shown in Figure 7. This was as expected from other studies involving rigid links. The 45° raster orientation inherently supports the hinge against bending. The lines of material at this raster angle behave as trusses angled at 45° to support a horizontal beam above or below it. The added strength of this raster angle becomes more evident in the larger loads. In Figure 4, it clearly supports more weight before deformation. In Figure 7, the deflection angles stand apart from the other two options, though for the 40% infill it takes a larger force to note this occurring.

The relative resistances of the other raster angles were also not affected by infill. The 0° raster orientation provides the next best support against bending because the force is orthogonal to the layers of material. The 90° raster orientation showed the least resistance to bending. This is because the orientation puts the microchannels in a position where they easily accept the deformation of the printed material. The joint then bends more easily.

One clear unintentional result found was that the plastic deformation limit was significantly influenced by the raster angle as shown in Figure 4. For the 0° raster orientation and 20% infill, elastic deformation occurred at 1.4175N. This was much sooner than at the other orientations which did allow for notably larger forces. The printing layers being parallel with the hinge do cause significant weakening of the joint. This is not entirely unexpected given that the layering position would cause significant tension on the top layer and compression on the bottom layer of the hinge. These layers do not have further inherent supports as with the other raster angles.

6. CONCLUSION

This paper shows that raster angle of compliant 3D printed hinges does affect the force-deflection relationship. Therefore, it is important to consider this when utilizing this method of manufacturing in soft robotics applications. A 45° raster angle provides the most resistance to deflection while the 90° raster angle provides the least. These results were consistent at both a 20% and a 40% infill. The larger infill did increase the resistance to deflection for each joint, however.

In the future, additional tests will be conducted on raster orientations to complete this phase of the project. More precise and detailed results will be obtained via the use of dynamic motion analysis in order to obtain data on the stress and strain as the joint moves. Investigations will also be done to determine if the raster angle affects the fatigue limit, as this is another important consideration in the selection and manufacturing of a joint to be used in a robotic application. Electronic actuation of the cable will be implemented so that this motion and cyclic testing may be performed on the joint.

As noted, other aspects of the cable actuation will be investigated as well including, but not limited to, joint shape and the routing of the cable within the hinge. This will create a full model for optimally modeling the joint for actuator selection and control design.

Eventually, multiple-DOF mechanisms will be investigated using these cable-driven soft joints. A tool for optimally designing cable-actuated soft robotics will be produced.

7. ACKNOWLEDGMENTS

We wish to acknowledge Edward Shynski, an undergraduate at Florida Polytechnic University, for his help in preparing and conducting some of the experimental aspects of this work.

8. REFERENCES

[1] J. S. Albus, R. V. Bolstelman, and N.G. Dagalakis, "The NIST robocrane," *J. Robotics System*, vol. 10, no.5, pp. 373-385, Sept. 1992.

[2] L. W. Tsai, "Design of tendon-driven manipulators," *J. Mechanical Design*, vol. 117, no. B, pp. 80–86, 1995, doi: 10.1115/1.2836474.

[3] M. Borgatti and K. Love, "Cable pull mechanisms – ready, willing, and cable," *Make: Soft Robotics*. San Francisco, CA: Maker Media, 2018, ch. 4, pp. 37-56.

[4] T. V. J. Tarvainen, W. Yu, and J. Gonzalez, "Development of MorphHand: design of an underactuated anthropomorphic rubber finger for a prosthetic hand using compliant joints," *IEEE Int. Conf. ROBIO*, 2012, doi: 10.1109/robio.2012.6490957.

[5] I. Hussain, Z. Iqbal, M. Malvezzi, L. Seneviratne, D. Gan, and D. Prattichizzo, "Modeling and prototyping of a soft prosthetic hand exploiting joint compliance and modularity," *IEEE Int. Conf. ROBIO*, 2018, doi: 10.1109/robio.2018.8665231.

[6] M. Tavakoli, A. Sayuk, J. Lourenço, and P. Neto, "Anthropomorphic finger for grasping applications: 3D printed endoskeleton in a soft skin," *Int. J. Adv. Manufacturing Tech.*, vol. 91, no. 5-8, 2017, pp. 2607–2620. doi: 10.1007/s00170-016-9971-8.

[7] R. Ozawa, K. Hashirii, and H. Kobayashi, "Design and control of underactuated tendon-driven mechanisms," *IEEE Int. Conf. Robotics and Automation*, 2009, doi: 10.1109/robot.2009.5152222.

[8] S. Shirafuji, S. Ikemoto, and K. Hosoda, "Design of an anthropomorphic tendon-driven robotic finger," *IEEE Int. Conf. ROBIO*, 2012, doi: 10.1109/robio.2012.6490995.

[9] D. T. Pham and R. S. Gault, "Solid ideas [rapid prototyping]," *Manufacturing Eng.*, vol. *75*, no. 5, pp. 239–243, Oct. 1996. doi: 10.1049/me:19960514.

[10] J. T. Cantrell, S. Rohde, D. Damiani, r. Gurnani, L. Disandro, J. Anton, and P. G. Ifju, P. G. "Experimental characterization of the mechanical properties of 3D-printed ABS and polycarbonate parts," *Rapid Prototyping J.*, vol. *23* no. 4, pp. 811–824, 2017, doi: 10.1108/rpj-03-2016-0042.

[11] L. Howell, *Compliant mechanisms*. New York: Wiley, 2002.

[12] P. Messimer, B. O'Toole, and M. Trabia, "Identification of the mechanical characteristics of 3D printed Ninjaflex," *Proceedings of the ASME Int. Mechanical Engineering Congress and Expo.*, 2019, IMECE2019-11674.

Compliant Joints Actuated Via Cables

Haghshenas-Jaryani, Mahdi
New Mexico State University
mahdihj@nmsu.edu

Estimated Utility of:
"Compliant Joint Actuated Via Cables"
Utility to Community: $20
Utility to Society: $20

Paper Purpose

Morris, Melissa
Embry-Riddle Aeronautical University
morrim44@erau.edu

Answer To Review:
"Compliant Joints Actuated Via Cables"
Utility Review: $20

The paper was adjusted to show data from more trials that were done, but not fully recorded before the draft was submitted. Data from more specimens have been added. It is true and intentional that the paper is for quasi-static analysis as dynamic analysis was not feasible at the time and will be done in the next stage now that proper equipment has been located. It is true that one shape was tested due to time and equipment limitations as well. This was only an initial effort and not a definitive work. We confirmed suspicions and findings taken from a different work in the context for this new use.

Interesting experimental paper

Wilde, Markus
Florida Institute of Technology
mwilde@fit.edu

Estimated Utility of:
"Compliant Joint Actuated Via Cables"
Utility to Community: $50
Utility to Society: $40

The paper presents the results of an experiment evaluating the bending stiffness of compliant robot manipulator joints made of 3D printed TPU. Three different raster angles are studied. The findings show significant differences in deflection curves for the three angles, showing trends that can be used in future joint designs. The topic is very relevant and the approach of 3D printing flexible joints for compliant robots is very interesting. The methodology and results of the tests are for the most part presented clearly.

The following elements of the paper should be considered more:

1. In Figure 4, the x axis is presented in grams. It would make more sense to use N, as it is the SI unit for forces and as it is used throughout the text.

2. It would be nice to have a photo or sketch of the complete test setup, in addition to the close ups. Right now, I assume that you attached calibrated weights to the end of the line dangling from the joint, but I can't be sure. The more detailed an experiment is explained, the easier to understand the results and the easier it is to reproduce your work.

3. There are many typos, grammatical errors and missing words that distract from the contents of the paper.

4. It would be cool if you used the test data to construct simple bending models for beams or trusses. Then you could use those models to inform future designs about the best raster angles to use. That way, you can start to fine tune the stiffness of joints to match requirements.

Corrected Errors

Morris, Melissa
Embry-Riddle Aeronautical University
morrim44@erau.edu

Answer To Review:
"Interesting experimental paper"
Utility Review: $40

Yes, the main point was the present that flexible robot joints could be 3D printed with cable actuation, but that a lot of work was left to be done.

1. The reviewer is correct that the proper dependent variable unit was not used. Figure 4 was corrected to show the force, and force was used in all subsequent graphs added as well for consistency.

2. Due to COVID19, it is impossible to access the set-up and take a photo of the entire thing. It is simply a sturdy jig with a clamp to hold the specimen. The write-up was clarified to better describe the set-up and included more details in that and the procedure.

3. The paper was revised and typos were corrected.

4. The suggestion to construct models is a good one. It is intended to use data and build models that can be used to design and plan these joints. However, that aspect is not yet done and is out of the timeframe and scope for this phase of the project.

Anomaly Detection in the Joint Angle Sensors of a Snake Robot

Luke Fina,* Hakki Erhan Sevil,† and Mahdi Haghshenas-Jaryani‡

ABSTRACT

Anomaly detection method for joint angle sensors of a snake robot is introduced. The main goal of this study is to detect the joint sensor with anomaly using statistical approach along 3 joint angle sensors in a four-link snake type platform. In the developed approach, each joint sensor's measurement is taken into account and χ^2 hypothesis test is adopted, and orthogonal projection method is used for pin-pointing the sensor that has the anomaly. The presented study is applied to a snake robot model in a computer simulation environment. Results and discussions are provided, and according to simulation results, developed model successfully detects and isolates joint angle sensor with anomaly. This study is a part of larger effort toward developing resilient and intelligent robotic systems.

Keywords: Snake Robot, Anomaly Detection, Joint Control.

1. Introduction

Biologically inspired robots have become again an emerging field in robotics due to the demand for developing autonomous robotic systems with capability to adapt and operate in unknown and dynamic environments. Especially, biological snake has fascinated roboticists over the past five decades due to their versatile limbless locomotion mechanism which adapt easily to unstructured and unknown environments [1]. Ever since preliminary study of biological snake locomotion by Gray in 1946 [2] and development of the first snake robot by Hirose [3] in early 70's, enormous amount of work have been focused on developing dynamic models and control algorithms for modeling and controlling these complex robotic systems [4].

Most of snake robots studied over the past 45 years were considered as a kinematic chain with non-holonomic constraints that were explicitly imposed for avoiding lateral slip (sideslip constraints); a crucial factor for generating the serpentine locomotion of snake robots which is the most common locomotion gait between biological snakes [4]. However, these constraints are unrealistic in practice, as in the case of biological snakes, they rely on the interaction of their body with surrounding environment based on anisotropic friction properties to generate these motion constraints and consequently the progressive motion. In this work, these explicit non-holonomic constraints are eliminated in order to develop a more realistic dynamic model of snake robots while a robust joint control algorithm is developed to modulate the internal body motion of the snake robot to produce the required serpentine locomotion by attenuating lateral slip at each link. This control algorithm is implemented on a planar four-link snake robot with 6 degrees-of-freedom.

Considering snake type robotic platforms, there are not many examples in the literature in terms of anomaly and/or fault detection methods. In one study, a comparison between simulation model and real-life experiments based approach is introduced to detect actuator faults [5]. In a different study, a failure recovery system is presented for modular robotic platforms [6]. Although failure recovery is presented, the assumption is that the fault/failure information is available, thus detection part is not developed nor applied. Similarly, bio-inspired fault adaptive system design is presented in the literature, without providing details or suggestions on how to detect anomalies or faults [7]. In this study, our aim is to introduce a statistical based anomaly detection system applied to joint angle sensors of a snake robot. It is very crucial for a snake type platform to have a fault free sensor measurements in order to successfully control its motion. This study is our first step to accomplish robust robotic snake platforms.

The remainder of the paper is organized as follows. The next section describes the dynamical system model of the snake robot. The joint control of the snake robot and developed anomaly detection algorithm are described in Section 3 and Section 4, respectively. The Section 5 provides information about simulation design, results, and discussions. In the final section, conclusions and proposed future work are presented.

2. Dynamic System Model

Dynamics of a planar four-link snake robot as shown in Fig. 1 is described in this section. This is based on our earlier work [8, 9] where the dynamic model of a general *n* link snake robot was developed. In contrast to most common model of snake robots which consider a series of non-holonomic constraints apply to links of the robot in order to avoid lateral movement (side-slip condition), these constraints were eliminated in our model to represent more realistic model of snake robots. That leads to an under-actuated dynamic system where the internal shape motion is not anymore directly related to the overall displacement of the snake robot. The free body diagram of a link is shown in Fig. 2. The reaction forces, $\mathbf{F}_i$, $\mathbf{F}_{i+1}$ and torque at the proximal and distal joints associated with the link, τ_i^z, τ_{i+1}^z as well as the friction forces, F_i^n, F_i^t, and torque,τ_i^b, applied at the center-of-mass of the link are shown here. Friction force were modeled as Coulomb friction and the friction torque as a linear damping terms related to the angular velocity of the link. The equations of motion were derived using Kane's method as pre-

*Undergraduate student, Department of Mechanical Engineering, University of West Florida, Pensacola, FL 32514

†Assistant Professor, Department of Intelligent Systems & Robotics, University of West Florida, Pensacola, FL, 32514

‡Assistant Professor, Department of Mechanical & Aerospace Engineering, New Mexico State University, Las Cruces, NM, 88003

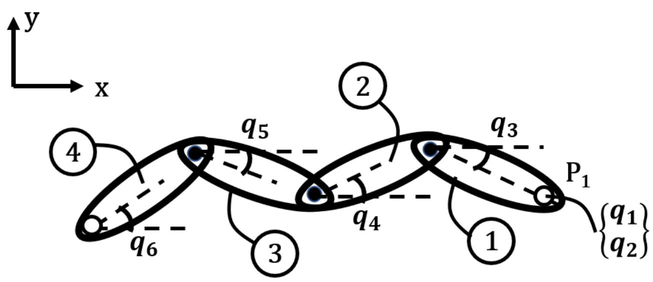


Figure 1. Overall Model of a Planar Snake Robot With Four Links and 3 Joints.

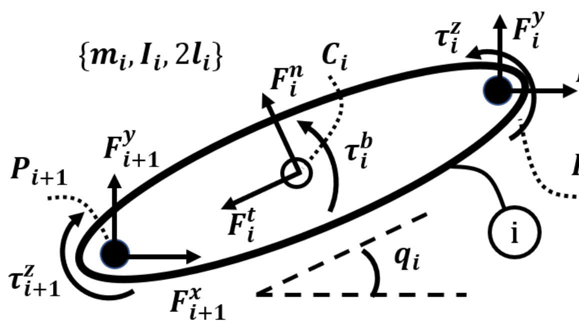


Figure 2. Free Body Diagram of a Snake Robot Link

sented in the general form of

$$M(\mathbf{q})\ddot{\mathbf{q}} + C(\mathbf{q},\dot{\mathbf{q}}) = J^T\mathbf{F} + B\tau \tag{1}$$

where the M is the symmetric and positive definite mass matrix, C is the nonlinear terms including centrifugal and Coriolis. External forces and moments acting on the snake robot (including friction forces and joint torques) are described by $\mathbf{F}$ and τ. $\mathbf{q} = [q_1, q_2, q_3, q_4, q_5, q_6]^T$ are the generalized coordinates describing the snake robot motion in 2D space where q_1 and q_2 are the coordinates of the tip of the snake robot (point P_1) and $[q_4, q_5, q_6, q_7]$ are the absolute rotational motion of the links with respect to the inertial frame O. $\dot{\mathbf{q}}$ and $\ddot{\mathbf{q}}$ are the first and second derivatives of the generalized coordinates, respectively. The equation of motion, Eq. (1), is derived using Kane's method where the differential equations are resultant of combination of two terms, the generalized inertial forces, F_r^*, and the generalized active forces, F_r, $r = \{1, \ldots, 6\}$ where

$$\begin{aligned} F_r^* &= -\sum_{i=1}^{4} (m_i \mathbf{a}_i \cdot \mathbf{V}_i^r + [\mathbf{I}]_i \alpha_i \cdot \omega_i^r) \\ F_r &= \sum_{i=1}^{4} (\mathbf{F}_i \cdot \mathbf{V}_i^r + \tau_i \cdot \omega_i^r) \end{aligned} \tag{2}$$

$\mathbf{V}_i^r$ and ω_i^r are linear and angular partial velocities, respectively, defined as follows;

$$\begin{aligned} \mathbf{V}_i^r &= \frac{\partial \mathbf{V}_i}{\partial \dot{\mathbf{q}}_r} = \frac{\partial \mathbf{V}_{P_i}}{\partial \dot{\mathbf{q}}_r} + \frac{\partial \omega_i}{\partial \dot{\mathbf{q}}_r} \times \mathbf{r}_{P_i C_i} \qquad (3) \\ &= \delta_{1,r}\hat{\mathbf{e}}_x + \delta_{2,r}\hat{\mathbf{e}}_y + \delta_{i+2,r}\hat{\mathbf{e}}_z \times \mathbf{r}_{P_i C_i} \\ \omega_i^r &= \frac{\partial \omega_i}{\partial \dot{\mathbf{q}}_r} = \delta_{i+2,r}\hat{\mathbf{e}}_z \qquad (4) \end{aligned}$$

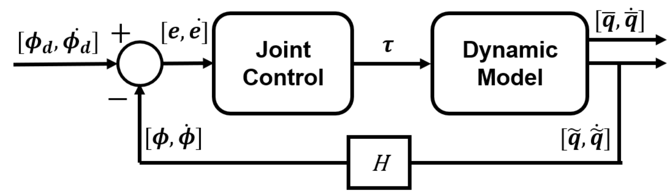


Figure 3. Joint Control Diagram.

where,

$$\delta i, j = \begin{cases} 1 & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases}$$

The active forces and moment acting on each body as shown in Fig. 2 are defined as

$$\begin{aligned} \mathbf{F}_i &= (F_i^x + F_{i+1}^x - \mu_n m_i g \ \mathrm{sgn}(V_i^n)\cos(q_i) - \\ &\quad \mu_t m_i g \ \mathrm{sgn}(V_i^t)\sin(q_i))\hat{\mathbf{e}}_x + \\ &\quad (F_i^y + F_{i+1}^y - \mu_n m_i g \ \mathrm{sgn}(V_i^n)\sin(q_i) - \\ &\quad \mu_t m_i g \ \mathrm{sgn}(V_i^t)\cos(q_i))\hat{\mathbf{e}}_y \\ \mathbf{M}_i &= \left(\tau_i^z + \tau_i^b - \tau_i^z\right)\hat{\mathbf{e}}_z \end{aligned} \tag{5}$$

and the inertial terms including linear and angular accelerations are defined as;

$$\begin{aligned} \mathbf{a}_i &= \ddot{q}_1\hat{\mathbf{e}}_x + \ddot{q}_2\hat{\mathbf{e}}_y + \alpha_i \times \mathbf{r}_{P_i C_i} + 2\omega_i \times \mathbf{V}_{P_i} \\ \alpha_i &= \ddot{q}_{i+2}\hat{\mathbf{e}}_z \end{aligned} \tag{6}$$

Substituting Eqs. (4, 5), and (6) back into Eq. (2) yields the equations of motion in Eq. (1). This set of equations will be used in the next section to derive the control algorithm in the joint space.

3. Joint Control of the Snake Robot

To generate the serpentine gait by the snake robot, the joints angle must be varying in a sinusoidal motion with amplitude α, phase shift of β, and bias γ as shown in Eq. (7)

$$\phi_i = \alpha \sin(\omega t + (i-1)\beta) + \gamma \tag{7}$$

To achieve this goal, a robust sliding mode controller is developed based on [8] for tracking control of joints of the snake-like robot in following the serpentine curve with parametric uncertainty as shown in Fig. 3. The forward dynamics of the snake robot provides the generalized coordinates and their derivatives after the numerical integration. The dynamic system's output, generalized coordinates, were split into the translational and rotational motion terms described by $\bar{\mathbf{q}} = [q_1, q_2]$ and $\tilde{\mathbf{q}} = [q_3, q_4, q_5, q_6]$, respectively. The rotational part was used to determine the joint motion defined by the relative angles between two adjacent links as follows;

$$\phi = H\tilde{\mathbf{q}} \quad \text{and} \quad \dot{\phi} = H\dot{\tilde{\mathbf{q}}} \tag{8}$$

where,

$$H_{3\times 4} = \begin{pmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{pmatrix}$$

Corresponding sliding mode manifold was defined based on the joint angle tracking error and its derivative

$$\mathbf{s} = \dot{\mathbf{e}} + \Lambda \mathbf{e} \tag{9}$$

where

$$\begin{aligned} \mathbf{e} &= \phi_d - \phi \\ \dot{\mathbf{e}} &= \dot{\phi}_d - \dot{\phi} \end{aligned} \tag{10}$$

The goal here is to design joint control action, τ, in order to the joint angles follow the desired sinusoidal trajectories presented in Eq. (7). Based on the sliding mode control approach, the control action is defined as

$$\tau = \tau_{eq} - K\text{sgn}(\mathbf{s}) \tag{11}$$

where τ_{eq} was determined by setting the time derivative of the sliding manifold in Eq. (9) equal zero and K is the diagonal matrix of control gains.

$$\dot{\mathbf{s}} = \ddot{\mathbf{e}} + \Lambda \dot{\mathbf{e}} = \mathbf{0} \tag{12}$$

Substituting from Eqs. (1) and (8) into Eq. (12), gives;

$$\begin{aligned} \tau_{eq} =& (H\tilde{M}^{-1}\hat{B})^{-1}\Big(\ddot{\phi}_d + \Lambda(\phi_d \\ & - H\dot{\bar{\mathbf{q}}}) - H\tilde{M}^{-1}(-C + J^T F)\Big) \end{aligned} \tag{13}$$

To prove the stability of the closed loop control and determining the control gain matrix K, the following Lyapunov candidate function was selected

$$V = \frac{1}{2}\mathbf{s}^T\mathbf{s} \tag{14}$$

with following condition

$$\dot{V} \leq -\eta \mid \mathbf{s} \mid \tag{15}$$

where η is positive. Using Eq. (14), the $\dot{V}$ will be determined as follows,

$$\begin{aligned} \dot{V} = \mathbf{s}^T\dot{\mathbf{s}} = \mathbf{s}^T\Big(\ddot{\phi}_d + \Lambda(\phi_d - H\dot{\bar{\mathbf{q}}}) - \\ H\tilde{M}^{-1}(-C + J^T F + B\tau)\Big) \end{aligned} \tag{16}$$

By substituting Eqs. (12) and (13) into Eq. (16) yields;

$$\begin{aligned} \mathbf{s}^T\Big(-\Delta_B(\ddot{\phi}_d + \Lambda(\phi_d - H\dot{\bar{\mathbf{q}}})) + \Delta_B H\tilde{M}^{-1}\hat{A} - \\ H\tilde{M}^{-1}\Delta_A\Big) + K \mid \mathbf{s} \mid \leq -\eta \mid \mathbf{s} \mid \end{aligned} \tag{17}$$

where

$$\begin{aligned} A &= -C(\dot{\mathbf{q}}, \mathbf{q}) + J^T F \\ \Delta_A &= A - \hat{A} \\ \Delta_B &= B\hat{B}^{-1} - I_B \end{aligned}$$

Thus, by chosing the control gain matrix K as follows the condition in Eq. (15) will be satisfied.

$$\begin{aligned} K = (H\tilde{M}^{-1}B)^{-1}\Big[\eta I + \Big(-\Delta_B(\ddot{\phi}_d + \Lambda(\phi_d - H\dot{\bar{\mathbf{q}}})) + \\ \Delta_B H\tilde{M}^{-1}\hat{A} - H\tilde{M}^{-1}\Delta_A\Big)\Big] \end{aligned} \tag{18}$$

4. Statistical Based Anomaly Detection

In this study, a statistical approach with chi-square test hypothesis is used for anomaly detection in the joint angle sensors of a snake robot[10, 11, 12, 13, 14]. The difference in measurement between different sensors is used to create residuals[15]. Three sensors at three joints of the snake robot are considered and the equations are provided correspondingly. The main assumption is that all three sensors have identical measurements with a phase shift, then the equation is written as

$$\begin{bmatrix} \phi_1(k) \\ \phi_2(k) \\ \phi_3(k) \end{bmatrix} = \begin{bmatrix} y(k) \\ y(k-k_1) \\ y(k-k_2) \end{bmatrix} + \begin{bmatrix} \xi_1(k) \\ \xi_2(k) \\ \xi_3(k) \end{bmatrix} + \begin{bmatrix} f_1(k) \\ f_2(k) \\ f_3(k) \end{bmatrix} \tag{19}$$

where $\phi_i(k)$ is the measured value from the i^{th} sensor of output state $y(k)$, $\xi_i(k)$ is the noise, and $f_i(k)$ is the fault value for the i^{th} sensor. The k_1 and k_2 are the phase shift values between sensor 1 and sensor 2, and between sensor 1 and sensor 3, respectively. The measurement noises $\xi_i(k), i = \{1,2,3\}$, are assumed to be normally distributed random sequences with zero mean and variance of σ^2 and independent from each other.

Equation (19) is pre-multiplied by the left null space of output state vector for eliminating the output state effect from the equation, then two-dimensional vector becomes

$$\mathrm{T}(k) = \mathbf{V}\mathrm{Y}(k) = \mathbf{V}\Xi(k) + \mathbf{V}\mathrm{F}(k) \tag{20}$$

where $\mathrm{T}(k) = [\tau_1(k)\ \tau_2(k)]^T \in \mathbb{R}^{2\times 1}$ is defined as the residuals, which are to indicate any anomaly in any sensor of the snake robot, $\phi_i(k)$, $i = \{1,2,3\}$.

In the normal case, the expected values of the residuals have a zero mean, on the other hand, if there is an anomaly, the expected value will be a non-zero value. This observation is used to set the hypothesis test for detecting anomaly as

$$\begin{aligned} H_0 &: E[\bar{\mathrm{T}}(k)] = 0 \\ H_1 &: E[\bar{\mathrm{T}}(k)] \neq 0 \end{aligned} \tag{21}$$

where the acceptance of the hypothesis is carried out by the second degree chi-square test as

$$\chi^2(2,k) = \gamma(k) = \bar{\tau}_1^2(k) + \bar{\tau}_2^2(k) \tag{22}$$

The χ^2 random variable $\gamma(k)$ is compared to h, which is a threshold value, for detecting the anomaly.

$$\Pr(\gamma(k) \leq h) = 1 - \alpha \tag{23}$$

If $\gamma(k)$ is equal or smaller than the threshold value, that means the system is normal. Contrary to that, if it's greater than threshold value, there is an anomaly on a sensor, with a false alarm probability of α. Once anomaly is detected, the isolation is performed by checking orthogonal projection of the fault vector on the null space[16], using ϑ as

$$\vartheta_i(k) = \left\| \frac{\bar{\mathrm{T}}(k)}{\|\bar{\mathrm{T}}(k)\|} - \frac{\mu_i}{\|\mu_i\|} \right\|, \ i = \{1,2,3\} \tag{24}$$

where μ_i is the normalized i^{th} column vector of the null space. Among $\vartheta_1(k)$, $\vartheta_2(k)$ and $\vartheta_3(k)$, the smallest $\vartheta_i(k)$ is

$$\vartheta_l(k) = \min\{\vartheta_i(k), i = 1,2,3\} \tag{25}$$

which will indicate the l^{th} sensor as the faulty one because the residuals will be shifted in the direction of the l^{th} column of the null space matrix. It should also be noted that, in this study, it is assumed that only one sensor has anomaly at a given time.

5. Results of Simulations and Discussions

The simulation experiments are accomplished by using 3 joint angle sensors in simulation environment. During the simulation runs, the snake robot is commanded to navigate in a straight line. The anomaly detection sub-system is designed to be just for providing information on anomaly, i.e. it is passive. In this study, the anomaly detection system is not integrated into the feedback control system; it runs as a separate unit.

The simulation consists of 4-link snake robot dynamic model, and the joint controller. In the simulations, the variance of sensor noise for joint angle measurements is set as 0.002 $(rad)^2$ [17]. The simulation is run for 30 seconds. In the resulting plots, the time starts from 5 seconds in order to eliminate the effect of different movements of the joints at the beginning of the simulation. The anomaly detection is based on the main assumption of the sensors are fault-free and identical. There is phase shift between different joints, and time delays are introduced for the sensors as, 0.17 seconds between sensor 1 and sensor 2, 0.33 seconds between sensor 1 and sensor 3. In the first simulation run, there is no anomaly added to the sensors.

The joint angle measurements are depicted in Fig. 4. The anomaly alarm index and anomaly sensor index plots are given in Fig. 5. The alarm index value gives 0 when there is no anomaly detected, and gives 1 when anomaly is detected. The sensor index provides the isolation information of the sensor with anomaly. It should be noted both alarm and sensor index results have false positives with probability α, which is set to 1%.

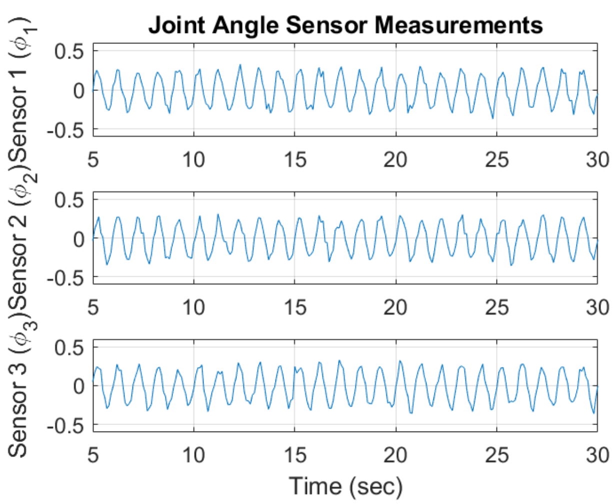


Figure 4. Joint Angle Measurements - No Fault Injected

As there is no anomaly existing in the sensors, orthogonal projection difference of the residuals scatter and they do not statistically favor any specific vector direction, meaning there is no anomaly detected. Moreover, alarm index and sensor index are presented

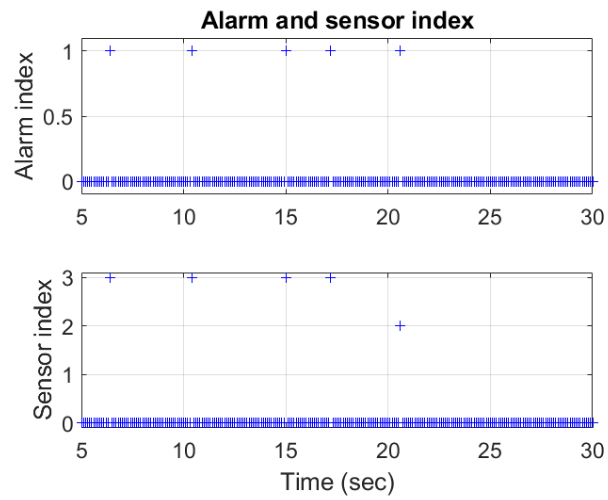


Figure 5. Alarm and Anomaly Index Values - No Fault Injected

in Fig. 5. The top plot shows the alarm index which equals to 1 whenever an anomaly is detected, and equals to zero otherwise. It can be seen that alarm index randomly gives value of 1 when there is no fault injected because of the false alarm probability, and total number of 1 values are less than 1%. The bottom plot in Fig. 5 depicts the sensor isolation index plot. The sensor index points out the sensor number that has the anomaly. Similarly, because of the false alarm probability, the sensor index gives 1, 2, or 3 in a random fashion even though no anomaly exists.

In the second simulation run, a fault is injected to the third sensor after 15 seconds in the simulation, for a duration of 10 seconds. The resulting sensor measurements are depicted in Fig. 6. Due to the injected anomaly, the third sensor has a biased type fault in its measurements. Furthermore, the anomaly values are given on Fig. 7.

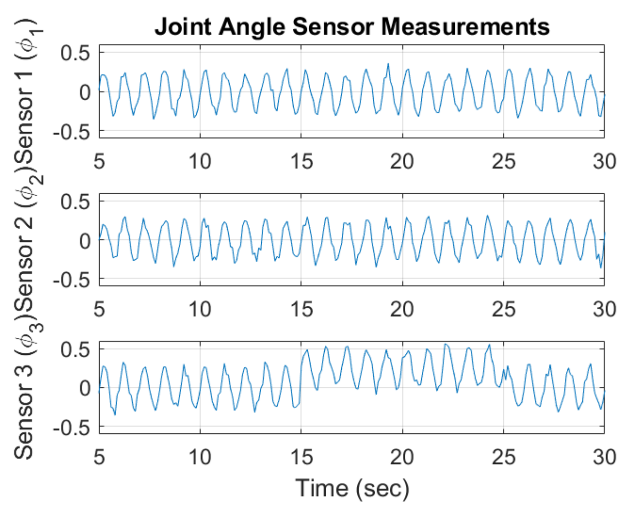


Figure 6. Joint Angle Measurements - Fault Injected

In the fault injected case, the developed method successfully detects the anomaly and pinpoints the sensor that has it (Fig. 7). In Fig. 7, alarm index and sensor index values are shown for joint angle measurements. During the anomaly occurrence from 15 to

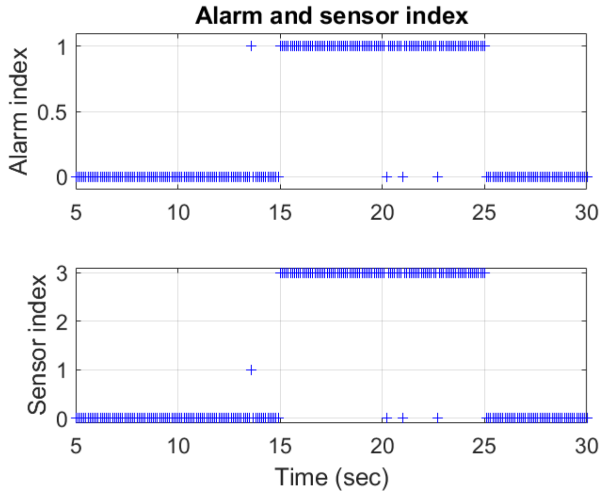


Figure 7. Alarm and Anomaly Index Values - Fault Injected

25 seconds of the simulation, the alarm index notifies the detection by giving value 1 consistently. Moreover, the sensor index marks sensor number 3 as the faulty sensor.

6. Conclusions

This paper introduces a anomaly detection method for joint angle sensors of a snake robot. For the detection part, the χ^2 hypothesis test is adopted, and isolation is performed by comparing orthogonal projections of three joint angle sensors. Simulation results are presented with successful implementation of the anomaly detection for an additive (bias) type fault in any of the three joint sensors. As future works, the developed system will be tested with different kind of faults, such as gain type, and parameter analysis will be added to determine robustness of the system considering noise levels and fault magnitudes. Furthermore, the effect of faulty sensor on the system dynamics will be discussed and robust fault tolerant controller design will be developed.

References

[1] K. Dowling, "Limbless locomotion: learning to crawl," in *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No. 99CH36288C)*, vol. 4, pp. 3001–3006, IEEE, 1999.

[2] J. Gray, "The mechanism of locomotion in snakes," *Journal of experimental biology*, vol. 23, no. 2, pp. 101–120, 1946.

[3] S. Hirose, *Biologically inspired robots: snake-like locomotors and manipulators*, vol. 1093. Oxford university press Oxford, 1993.

[4] P. Liljebäck, K. Y. Pettersen, . Stavdahl, and J. T. Gravdahl, "A review on modelling, implementation, and control of snake robots," *Robotics and Autonomous Systems*, vol. 60, pp. 29–40, Jan. 2012.

[5] V. Mehta, S. Brennan, and F. Gandhi, "Experimentally verified optimal serpentine gait and hyperredundancy of a rigid-link snake robot," *IEEE Transactions on Robotics*, vol. 24, no. 2, pp. 348–360, 2008.

[6] V. Vonásek, D. Oertel, S. Neumann, and H. Wörn, "Failure recovery for modular robot movements without reassembling modules," in *2015 10th International Workshop on Robot Motion and Control (RoMoCo)*, pp. 136–141, IEEE, 2015.

[7] M. Arroyo, N. Huisman, and D. C. Jensen, "Exploring natural strategies for bio-inspired fault adaptive systems design," *Journal of Mechanical Design*, vol. 140, no. 9, p. 091101, 2018.

[8] M. Haghshenas-Jaryani and G. Vossoughi, "Modeling and sliding mode control of a snake-like robot with holonomic constraints," in *2008 IEEE International Conference on Robotics and Biomimetics*, pp. 454–461, 2009.

[9] M. Haghshenas-Jaryani and G. Vossoughi, "Trajectory control of snake-like robots in operational space using a double layer sliding mode controller," in *ASME 2015 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, pp. V05AT08A053–V05AT08A053, American Society of Mechanical Engineers, 2015.

[10] H. E. Sevil and A. Dogan, "False fault detection in airdata sensor due to nonuniform wind in aerial refueling," in *AIAA Atmospheric Flight Mechanics Conference*, p. 6446, 2011.

[11] H. E. Sevil, "Airdata sensor based position estimation and fault diagnosis in aerial refueling," phd dissertation, The University of Texas at Arlington, Arlington, TX, USA, 2013.

[12] H. E. Sevil and A. Dogan, "Airdata sensor fault detection and isolation for receiver aircraft in aerial refueling," in *Proc. of ASM 2013, AIAA Aerospace Sciences Meeting*, (Grapevine, USA), 7-10 January 2013. AIAA 2013-0950.

[13] H. E. Sevil and A. Dogan, "Fault diagnosis in air data sensors for receiver aircraft in aerial refueling," *AIAA Journal of Guidance, Control and Dynamics*, vol. 38, no. 10, pp. 1959–1975.

[14] H. E. Sevil, "Anomaly detection using parity space approach in team of UAVs with entropy based distributed behavior," in *Proc. of AIAA Scitech 2020 Forum*, (Orlando, USA), 6-10 January 2020. AIAA 2020-1625.

[15] I. Hwang, S. Kim, Y. Kim, and C. E. Seah, "A survey of fault detection, isolation, and reconfiguration methods," *IEEE transactions on control systems technology*, vol. 18, no. 3, pp. 636–653, 2010.

[16] P. M. Frank, "Fault diagnosis in dynamic systems using analytical and knowledge-based redundancy: A survey and some new results," *automatica*, vol. 26, no. 3, pp. 459–474, 1990.

[17] H. E. Sevil and A. Dogan, "Investigation of measurement noise effect on wind field estimation using multiple UAVs," in *Proc. of AIAA Scitech 2019 Forum*, (San Diego, USA), 7-11 January 2019. AIAA 2019-1601.

Review

Morris, Melissa
Embry-Riddle Aeronautical University
morrim44@erau.edu

Estimated Utility of:
"Anomaly Detection in the Joint Angle Sensors of a Snake Robot"
Utility to Community: \$80
Utility to Society: \$60

In summary, the paper argues that a statistical method can be used to detect error in a sensor of a snake robot. It's claimed contributions are:

The paper is interesting and shows very interesting results. It would be more robust to show more trials with different kinds of errors, but it is clear that the method works for the instance given where a sensor has a notable drift from others. The entire planned research is very interesting and will be a great contribution. This is a good first step toward it!

Response to review - Anomaly Detection for Snake Robot

Sevil, Hakki Erhan
University of West Florida
`hsevil@uwf.edu`

Answer To Review:
“Review”
Utility Review: $50

Patient Walker Application and Human Subject Tests with an Assistive Robot

Luke Fina,[*] Cody Lee Lundberg,[†] Hakki Erhan Sevil,[‡] Deborah Behan,[§] and Dan O. Popa[¶]

ABSTRACT

In this study, we demonstrate a proof of concept of using an off the shelf robot, PR2 (Personal Robot 2), as nursing assistant to perform a particular task, that is assisting patients with walker. The robotic platform used in this project (PR2) has autonomous navigation and obstacle avoidance capabilities. An android tablet running ROS (Robot Operating System) Java is developed for the users to interact with PR2. The experiments are performed in Assistive Robotics Laboratory in the University of Texas at Arlington Research Institute (UTARI). In the laboratory, a real-world hospital setup is designed for this study, with a patient bed and walker. Patient walker is stationed in fixed location for PR2 to navigate easily. We successfully demonstrated patient walker task, and human subject tests are performed with volunteers and results are presented.
Keywords: Assistive Robotics, Service Robotics, Human-Robot Interaction.

1. Introduction

Patients with disabilities and with less mobility, often require one-to-one assistance to manage their daily activities. Due to increase in number of patients, the nurses are not able to offer enough care and attention to patients [1]. By using robotic assistants for nursing tasks, we can free up some of the nurses time and prioritize their tasks with patients who have severe health conditions. In literature, there are robotic systems which are developed to help patients with activities of their daily living without needing much help from others. "XR4000", a walker robot [2], with inbuilt walker, assists elderly people to walk to a destination autonomously using a pre-defined map. The PAMM (Personal Aid for Mobility and Monitoring) robot discussed in [3] adds additional functionalities like obstacle avoidance and navigation guidance to existing walkers. When using the robot, it can monitor health condition of the users and informs caregivers in case if any emergency situation is detected.

Based on their surroundings, people tend to walk with different speeds. To assist the user in such situations, robotic platforms have to change their speed with respect to the user actions. An omnidirectional moving cane robot is discussed in [4, 5] uses information from various sensors like force, laser, and tilt sensors to predict user's actions. Based on the prediction, the robot adjusts its speed and facilitates users to move at variable speed depending on the situation. The PAM-AID (Personal Adaptive Mobility) robot discussed in [34] detects surroundings and delivers the information using speech to the users, helping the blind people to navigate and interact with the surroundings. A similar technique is used in our current project to assist patients with the walker. The PR2 supports users and prevents them from falling, similar to the robot discussed in [6].

In this study, we build upon our previous studies [7, 8] and add patient walker task, which consists of assisting patients with the walker. "Fetch Cart", "Start Walker" and "Stop Walker" functions on the android tablet are used for this task. When the user requests "Fetch cart", PR2 fetches the walker to user's bed. A novel Multi Model Proportional (MMP) controller path planning technique is used to navigate the walker back to user's bed since walker blocks PR2's LIDAR sensors. This technique enables PR2 to navigate a linear path instead of a curved path. Once patient is ready to use the walker, he/she selects "Start Walker" from the android tablet and starts walking with the PR2. In this mode, PR2 becomes less rigid and enables the user to guide it. By using changes in arm positions, PR2 detects which motion user is going to take and issues commands to its base controller to follow the user with the same motion. Using this technique, PR2 walks with the user and supports him/her to a destination. Once the patient reaches the destination, he/she selects "Stop Walker" function on the android tablet and, PR2 regains its rigid functionality again and no longer walks with the patient.

The remainder of the paper is organized as follows. The next section describes the developed algorithm for patient walker task. The details about patient walker hardware and parameter analysis are described in Section 3 and Section 4, respectively. The Section 5 provides information about human subject test design, results, and discussions. In Section 6, conclusions and proposed future work are presented.

2. Patient Walker Algorithm

The patient walker task involves multiple forms of autonomous navigation. The robot uses of the ROS navigation stack and 2DNav [9] method for navigating in dynamic cluttered environments full of obstacles. ROS 2DNav [9] is designed to flatten the robot and environment geometry into a two dimensional plane for path planning and obstacle avoidance. This works well with small objects being carried by the robot's grippers but will fail if the robot needs to move a larger object. In a hospital like environment, when the robot

[*]Undergraduate student, Department of Mechanical Engineering, University of West Florida, Pensacola, FL 32514

[†]Research Engineer II, The University of Texas at Arlington Research Institute (UTARI), Fort Worth, TX, 76118

[‡]Assistant Professor, Department of Intelligent Systems & Robotics, University of West Florida, Pensacola, FL, 32514

[§]Clinical Associate Professor, College of Nursing and Health Innovation, The University of Texas at Arlington, TX, 76019

[¶]Professor, Electrical & Computer Engineering Department, The University of Louisville, Louisville, KY 40292

Table 1. Parameter Analysis Results

	Input					Output										
	V_{limit}	W_{limit}	P_x	P_y	P_w	F_{lmax}	F_{rmax}	F_{lmin}	F_{rmin}	F_{lmean}	F_{rmean}	F_{lvar}	F_{rvar}	V_{max}	V_{mean}	V_{var}
Case 1	0.35	0.45	0.30	0.30	2	30.17	51.56	10.40	11.33	13.70	27.29	1.20	5.59	0.35	0.14	0.02
Case 2	0.30	0.45	0.30	0.30	2	23.88	42.20	12.27	16.18	16.04	25.79	0.98	4.14	0.30	0.14	0.01
Case 3	0.25	0.60	0.30	0.30	2	23.17	45.23	10.55	17.25	13.77	27.63	0.83	4.42	0.24	0.13	0.01
Case 4	0.25	0.52	0.30	0.30	2	21.72	41.99	11.94	20.01	15.20	26.83	0.76	6.66	0.24	0.13	0.01
Case 5	0.25	0.45	0.60	0.30	2	28.16	42.50	10.89	16.31	13.77	27.57	0.60	4.01	0.24	0.17	0.01
Case 6	0.25	0.45	0.45	0.30	2	23.58	38.46	11.49	13.90	15.86	24.99	0.98	3.09	0.24	0.15	0.01
Case 7	0.25	0.45	0.30	0.60	2	22.77	45.35	10.24	16.26	13.37	27.81	0.83	4.50	0.24	0.13	0.01
Case 8	0.25	0.45	0.30	0.42	2	22.83	45.60	10.25	17.69	14.53	26.83	1.17	8.31	0.24	0.13	0.01
Case 9	0.25	0.45	0.30	0.30	2.50	22.95	44.36	10.58	18.96	13.37	28.12	0.83	5	0.24	0.13	0.01
Case 10	0.25	0.45	0.30	0.30	2.25	22.78	41.18	11.87	17.32	15.76	25.07	1.10	3.94	0.24	0.13	0.01
Case 11	0.25	0.45	0.30	0.30	2	23.18	43.94	9.92	14.74	13.75	27.89	1.19	5.42	0.24	0.13	0.01

needs to move a walker, or an IV pole, it affects the algorithms ability to flatten and separate the carried items from the robot's base and dynamic environment. As a solution, we expand the flattened robot footprint include the area occupied by either the IV pole or walker. This helps with both defining the object as being rigidly attached to the robot and avoiding collisions between the carried object and environment. For safety during the human trials, however, this method was not used. While pushing the walker back to the patient's bed a simpler multimodal proportional controller is used. A diagram of this controller is shown in Fig. 1. One of the goals of this controller is for the robot to be perceived by user as to move predictably, smooth, and to move a cart, walker, or IV pole safely.

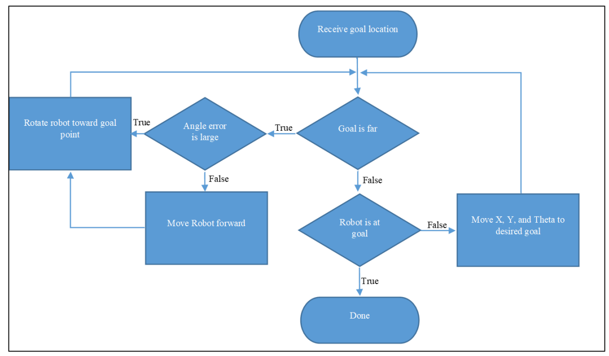


Figure 1. Patient Walker Control Algorithm Diagram

Since the robot follows and supports user motion in this task, robot's reactions with the walker should be smooth and easy to operate. In this task, when user applies force on grippers and leads robot to a desired location, the robot should detect user's intentions to move along with the user. Previously developed PID controller uses traditional force based logic that tries to maintain a desired force all the time during the motion. Since, our experiment requires operating at variable speeds, using that PID controller is not suitable for this task. Also, the PID controller can become highly unstable when working along with users. A novel impedance controller based technique called, 'Compliance controller' or 'Stiffness controller' has been developed specially for this task [10]. When user selects 'Start Walker' function on the android tablet, this controller gets initialized by PR2. In this mode, PR2 loses 3 DOF (Degrees Of Freedom) and motion only lies in walker's X, Y, and Z planes. Two hyper-parameters, a task position that is in front of PR2, and a stiffness force parameter are set before starting experiment, for the compliance controller. When user applies force greater than that off the stiffness controller, PR2 grippers move freely to a new position, and changes the pre-defined task coordinates of grippers. This motion creates an error in task space, to minimize this error, PR2 drives its base and grippers close to the home pose. This technique is used by the PR2 to coordinate and move along with the patient walker. This motion is continued until the patient selects "Stop Walker" functionality on the android tablet. During this application, to create smooth operation of PR2 with users, velocity gain of PR2 is set to low in the X, Y, and Z directions.

In walking mode, PR2 uses a compliance controller to move the arms to a desired position relative to the torso while holding onto the walker. When the arms reach a relatively low stiffness threshold and can no longer hold the arms in the desired position, the robot will move the base, using a PD controller based on the error of the gripper positions. This controller allows the robot to follow the walker while applying a directionally adjustable level of stiffness to the walker for stability. Additional improvements can be made to both the walker retrieval and walking modes. While the robot pushes the walker to the patient, it uses 2DNav [9] with an expanded footprint for more efficiency, and if additional sensors were added to the robot, that would compensate for the primary LIDAR being obscured. On a custom robot, the sensor could be place at a location less likely to be obscured.

The walking mode allows the patient to adjust the stiffness the arms use to hold the walker into position. This would allow different patients to use the walking mode more comfortably with different settings. The stiffness can also automatically be adjusted based on the stability of the patient but this would require the robot to be able to measure the walking stability of the patient. One method would be to use the face tracker already operating on the robot and the acceleration the patient applies to the walker. If the patient begins to fall, this would be measured through a rapid acceleration and a lowering of the face. The robot would rapidly increase the stiffness of its arms to stabilize the patient and help preventing a fall. Further details about the patient walker algorithm can be found in [10].

3. Details of Modified Patient Walker

In this study, we use "Drive Medical Walker HX5 9JP" model no. 10226-1 for patient walker experiments. The 4 wheeled feet provides easy steering, the aluminum build makes the walker light weight, and requires less effort to walk. The walker can hold up to 350 lbs and has dimensions of 16.75" x 25" (Length x Width), and comes with 5' wheels. It provides easy mobility for people with disabilities, and elderly people. The walker is modified with a handle to support PR2 robot grippers to hold it, and a tray is added to place tablet during experiments. The final design of walker is shown in the Fig. 2.

In order for the patient to be able to rotate at their centroid, the centroid of the robot, or their shared centroid, the walker was modified to have four caster wheels. In a traditional setting, the extra caster wheels could reduce the stability granted by the walker, but in this case the robot is used to increase stability for the patient. The casters allow the robot to make use of its dexterous holonomic base and allow the patient to choose between multiple paths to reach the same goal position. Added handle, which is made of a piece of PVC pipe, is used for the robot to grip and stabilize the walker. The added tray is located at the top of the walker to securely hold the patient's tablet during navigation, and it consists of a slot that the tablets fits into and Velcro.

Figure 2. Modified Patient Walker

4. Parameter Analysis

A parameter analysis is performed with 11 Cases to optimize the robot navigation while retrieving the walker. The analysis is preformed along the preferred path the robot has access to during the human subject testing. The following parameters are varied: the maximum linear velocity (V_{limit}), the maximum angular velocity (W_{limit}), the forward proportional gain (P_x), the sideward proportional gain (P_y), and the angular proportional gain (P_w). These cases are listed with the tested values in Table 1.

The analysis is evaluated by considering 4 force values and 3 velocity values. These values are: the maximum recorded force (F_{max}), the minimum recorded force (F_{min}), the mean of the recorded force (F_{mean}), and variance of the recorded force (F_{var}), the maximum recorded velocity, the mean of the recorded velocity, and the variance of the recorded velocity. The force values can be separated between the left gripper and right gripper such as F_{lmax}, and F_{rmax}.

The input parameters in Case 11 are used during the human subject testing. These parameters are chosen in order to reduce the maximum force measured in both grippers, to contact the walker handle without pushing it out of the open grippers before grasping, complete the experiment in a timely manner, and move the robot without aggressive maneuvers. Case 11 does not have the lowest force for either gripper but keeps the force in both grippers low without raising the force of the opposing gripper, while not increasing the angular speed of the robot. The values in Case 11 are more likely to be able to have the grippers contact the walker handle to grasp it without pushing the handle out of the gripper.

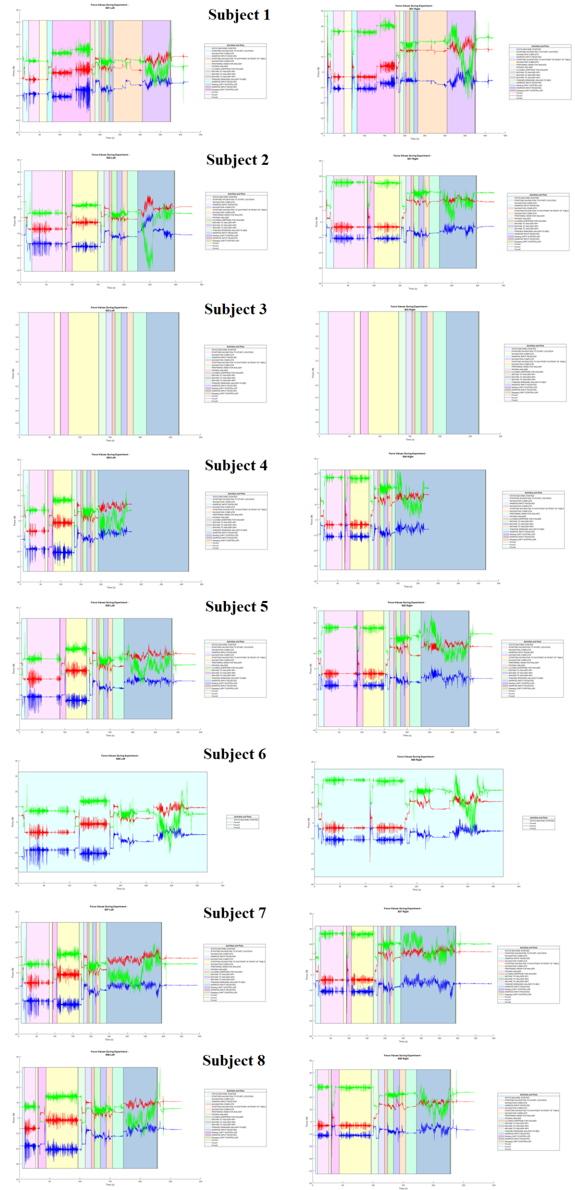


Figure 3. The output plots of the force sensors - Subject 1-8 - Subject 3 force data was lost after recording due to file corruption. Only demo phase timing data exists - Subject 6 demo phase timing data was lost after recording due to file corruption. Only force torque data exists.

5. Human Subject Tests and Results

Human subject tests are performed for the patient walker task with a total of 8 volunteers. Details of human subject tests for patient walker task are given below.

The patient walker task begins with the patient in a bed and having access to a tablet to communicate with the robot. A customized walker is stored at a separate location. When the demo starts, the robot will prepare by tucking its arms and adjusting the height of its torso then navigate to the bedside. The robot will then wait for a command from the patient while occasionally reminding them verbally to use the tablet to interact.

When the patient selects walker task on the tablet the robot will then navigate to retrieve the walker using the ROS 2DNav algorithm [9]. When the robot is positioned in front of the walker, it untuck its arms and places them into the gripping position. The multimodal proportional controller is used to contact the walker. The robot closes its grippers and uses the multimodal proportional controller to gently push the walker to the patient's bed. The patient can then stand up, place the tablet onto the walker, and use the tablet to start the walking mode on the robot. The patient can then push and pull the walker in any horizontal direction. The robot will sense the motion of the walker and follow it, while limiting the walkers speed for stability. When the patient has arrived at their desired location, he/she can stop the walking mode and the robot will hold the walker rigidly in place. An snapshot from a test run is depicted in Fig.4.

Human subject tests for patient walker task are performed in Assistive Robotics Laboratory (ASL) at UTARI. 8 volunteers come to ASL for human subject tests, and some of the volunteers are actual Registered Nurses (RNs). The output plots of the force sensors are presented in Fig.3. The comments from volunteers and observations during the human subject tests are given below; which are provided as recommendations for development of custom ARNA platforms.

Observations:

- Patient can't be sure when to press button (Test 1).
- PR2 has hard time to navigate to the walker (Test 1).
- PR2 has hard time to find walker (Test 1).
- One of the grippers misses the walker handle (Test 1).
- Patient says turning is tricky (Test 1).
- Patient forgets to turn off the walker mode (Test 4).
- Initialization is failed, experiment is started over (Test 5).
- During navigation to the walker, PR2 is failed. Experiment is restarted (Test 5).
- During navigation to the walker, PR2 is failed again. Experiment is restarted (Test 5).
- Patient says that the rotation is hard and tricky (Test 6).

Figure 4. Snapshot of a Test for Patient Walker Task

6. Conclusions

In this study, we present outcomes of nursing assistant tasks using an assistive robot (PR2). Our main focus is to implement patient walker task for assisting patients with walking activity. Parameter analysis is performed and parameters with best results are selected to be used in the human subject tests. Human subject tests are performed with 8 volunteers. In the experiments with human subjects, in all cases algorithm works successfully in assisting volunteer's walking. The hardware and software parts of the system are open to advance improvements.

7. Acknowledgments

This work was supported by the National Science Foundation (NSF) Partnerships for Innovation: Building Innovation Capacity (PFI: BIC) grant (Award Number: 1643989).

References

[1] The Wall Street journal, "Nurses Shift, Aiming for More Time with Patients," URL: `https://www.wsj.com/articles/nurses-shift-aiming-for-more-time-\with-patients-1405984193` (visited on July 3, 2018).

[2] A. Morris, R. Donamukkala, A. Kapuria, A. Steinfeld, J. T. Matthews, J. Dunbar-Jacob, and S. Thrun, "A robotic walker that provides guidance," in *2003 IEEE International Conference on Robotics and Automation (Cat. No. 03CH37422)*, vol. 1, pp. 25–30, IEEE, 2003.

[3] S. Dubowsky, F. Genot, S. Godding, H. Kozono, A. Skwersky, H. Yu, and L. S. Yu, "Pamm-a robotic aid to the elderly for mobility assistance and monitoring: a" helping-hand" for the elderly," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, vol. 1, pp. 570–576, IEEE, 2000.

[4] K. Wakita, J. Huang, P. Di, K. Sekiyama, and T. Fukuda, "Human-walking-intention-based motion control of an omnidirectional-type cane robot," *IEEE/ASME Transactions On Mechatronics*, vol. 18, no. 1, pp. 285–296, 2011.

[5] H. Wang, B. Sun, X. Wu, H. Wang, and Z. Tang, "An intelligent cane walker robot based on force control," in *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 1333–1337, IEEE, 2015.

[6] J. Huang, P. Di, T. Fukuda, and T. Matsuno, "Motion control of omni-directional type cane robot based on human intention," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 273–278, IEEE, 2008.

[7] A. V. Dalal, A. M. Ghadge, C. L. Lundberg, J. Shin, H. E. Sevil, D. Behan, and D. O. Popa, "Implementation of object fetching task and human subject tests using an assistive robot," in *Proc. of ASME 2017 International Mechanical Engineering Congress and Exposition (IMECE2017)*, (Atlanta, USA), 30 September-3 October 2018. DSCC2018-9248.

[8] A. M. Ghadge, A. V. Dalal, C. L. Lundberg, H. E. Sevil, D. Behan, and D. O. Popa, "Robotic nursing assistants: Human temperature measurement case study," in *Proc. of 32nd Florida Conference on Recent Advances in Robotics FCRAR*, (Lakeland, USA), 9-10 May 2019.

[9] ROS Wiki, "navigation Package Summary," URL: `http://wiki.ros.org/navigation` (visited on April 10, 2018).

[10] S. Cremer, I. Ranatunga, and D. O. Popa, "Robotic waiter with physical co-manipulation capabilities," in *2014 IEEE International Conference on Automation Science and Engineering (CASE)*, pp. 1153–1158, IEEE, 2014.

Feedback on patient walker with assistive robot

Silaghi-Horisberger, Anne Melody
UNIL
amelodysilaghi@hotmail.com

Estimated Utility of:
"Patient Walker Application and Human Subject Tests with an Assistive Robot"
Utility to Community: $100000
Utility to Society: $100000

The authors tested the tasks of a walker attached to an off the shelf robot (PR2). The robot assists patients by bringing the walker to the bed side, sustaining/supervising the walk of the patient holding the walker while moving to a predefined place, and stopping the walker upon arrival.

Extending on their prior work, the authors first modified an existing walker to improve as much as possible its usability. Then, they optimized the parameters of the robot navigation to improve the success rate for the action where the robot grips the walker. Finally, they studied the walker task with 8 human subjects, and analyzed the output plots of the force sensors, as well as the observations of the volunteers.

In all cases, the robot-walker pair assisted successfully the volunteer's walking.

The article is generally well written.

Response to review - Patient walker tests with assistive robot

Sevil, Hakki Erhan
University of West Florida
hsevil@uwf.edu

Answer To Review:
"Feedback on patient walker with assistive robot"
Utility Review: $50

Cold Spray Additive Manufacturing - Process Review

Kumar Yogesh Shah

Mechanical and Materials Engineering Department
Florida International University
10555 W. Flagler St. Miami, FL, 33174, USA

kushah@fiu.edu

ABSTRACT
The field of Additive Manufacturing (AM) has come a long way since the nineteen-eighties when it was first introduced. One of the newest AM techniques is Cold Spray Additive Manufacturing (CSAM). This method is a relatively new research field for researchers and industries worldwide. The processes of Cold Spray, CSAM, and its key aspects will be discussed. These topics include discussions regarding the growth of this technology, Cold Spray fundamentals, parameters, and processes, CSAM pre and post processing analysis features, and newest research areas this process has been implemented in. Findings from this paper are intended to fully review this emerging technology to a fundamental understanding.

Keywords
Cold Spray (CS), Additive Manufacturing (AM), Cold Spray Additive Manufacturing (CSAM), Solid-State Manufacturing, Deposition Mechanism, Manufacturing Strategy

1. INTRODUCTION
Additive Manufacturing (AM) was introduced in the 1980s [1]. Initially creating basic plastic parts with huge errors, to currently printing parts made from high performance plastics, metals, and composites with extremely accurate tolerances. In the last decade this field has moved from only making prototyping parts to industrial applications. The term AM means; manufacturing is occurring by the addition of material in order to form a desired part.

Standard AM techniques, such as Fused Deposition Modeling (FDM), are performed in which plastics are thermally extruded from a nozzle and selectively deposited layer-by-layer [2]. Newer methods have been developed by companies such as Markforged with their Metal X Printers. These advanced printers use Atomic Diffusion Additive Manufacturing (ADAM), in which they create metal parts using plastic spooled filaments enlaced with fine metal powders [3]. The filament is used to manufacture parts using FDM, after which they are sintered at high temperatures to cause atomic diffusion [3].

Cold Spray Additive Manufacturing (CSAM) is the newest AM technique that researchers and industries are developing. This method is a state-of-the-art process originally developed from application techniques for coating surfaces [4]. Fine metal powders are accelerated in a system to supersonic velocities in order to be imbedded onto a substrate or surface [5]. These particles of powder are imbedded in the same manner as traditional AM, layer-by-layer, or so called, particle-by-particle. Unlike most AM processes, the materials used in CSAM do not undergo a thermal deformation, melting, or diffusion from its original state [6]. This enables a variety of desired features from parts created through CSAM.

The topic of Cold Spray has over 2000 research papers written about it, over the last twenty years. CSAM, being a relatively new topics of research, only has over 30 papers, as indicated in Fig. 1. This graph shows the quantity of research papers published over the past twenty years using two well-known search engines. As demonstrated by the flow of data, there is an upwards trend of research being conducted in the area of CSAM, so far peaking last year at 47% of total papers. This trend occurs due to the desired ability of industries to partake in this process to create better parts. The following paper will discuss and review the processes and key aspects of Cold Spray focusing on CSAM.

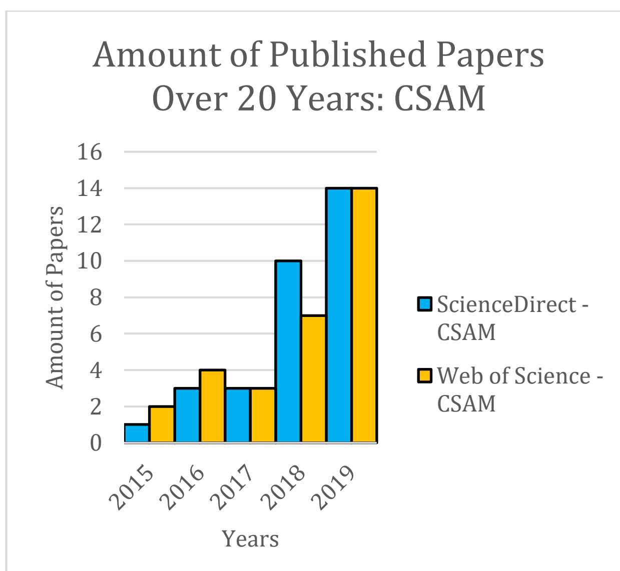


Figure 1: Quantity of research over the past twenty years: CSAM

2. COLD SPRAY

Cold Spray was developed in the 1980s around the same time AM was being developed [1]. This method was to aid in the manufacturing of parts with unique enhanced coatings [4]. These surface coatings enabled an extra layer of protection to metal parts adding beneficial properties to the overall part. Not only was it used in the enhancement of surface coatings, but it allows for a new method of repair for existing surfaces [6]. These damaged parts contain cracks and defects that can lead to mechanical failures [6]. By using Cold Spray to potentially fix these parts in turn allows for industries such as aerospace, automotive, and the military to create more serviceable systems; instead of resorting to the replacement of parts [6]. This allows these industries to use state-of-the-art methods and to spend less resources on critical repairs.

As it is known, Cold Spray is a solid-state process in which materials of fine metal powders are imbedded onto a substrate without changing properties from original feedstock [5]. These particles are accelerated with a highly compressed gas to be supersonically deposited onto a surface at velocities in excess of 300 m/s [5]. The most ideal gas for this process is helium, but due to high production costs, nitrogen, air, or a mix of them is mostly used instead [7]. Since these particles will not experience any melting, adherence of particles to the substrate and to each other, as the layers build, is due to localized plastic deformation [8]. Due to this deformation, each particle experiences mechanical interlocking and sometimes metallurgical bounding during deposition [7]. This in turn allows for the reduction of oxidation, phase transformations, defects, and residual stresses within the deposited piece [7-9].

The novel Cold Spray system was developed into two versions, low pressure and high pressure. Each of these systems has its benefits and applications that can be schematically laid out as shown in Fig. 2. In the more common high pressure system, a compressed gas is used in both the heating and powder feeding streams, after which they converge in the de-Laval convergent-divergent nozzle to be sprayed [7]. In the cheaper, portable, low pressure system, an air compressor serves as the compressed gas source, which is heated and meets the powder directly in the nozzle for spraying [5]. For applications needing to achieve a denser, thicker coating, a high pressure system is recommended [8]. In cases were portable coatings or repair is needed on delicate parts a low pressure system is more feasible [5].

Cold Spray encompasses a variety of parameters and aspects to allow the system to run and deposit materials in a uniform manner. As this machine relies on relatively high velocities to ensure proper deposition, one key term when controlling the process is critical impact velocity [10-11]. This velocity is the specific speed a particle needs to travel as it impacts the surface of the substrate, to properly adhere to that surface fully [12]. Since a variety of materials can be sprayed, this critical velocity is not a constant parameter but fluctuates depending on a variety of components such as particle type, size, and temperature [10].

Particle powders are mostly made up of the softer metals like aluminum and copper, but extend into categories as follows: Metals (Ni, Ti, Ag, Zn, Ta, Nb), Refractory metals (Zr, W, Ta), Alloys (steels, Ni alloys, MCrAlYs, Al-alloys), and Composites (Cu-W, Al-SiC, Al-Al2O3) [13]. The particle sizing of these fine metal powders must consist of diameters typically between 10 and 100 microns, to ensure a high velocity, thus leading to a complete

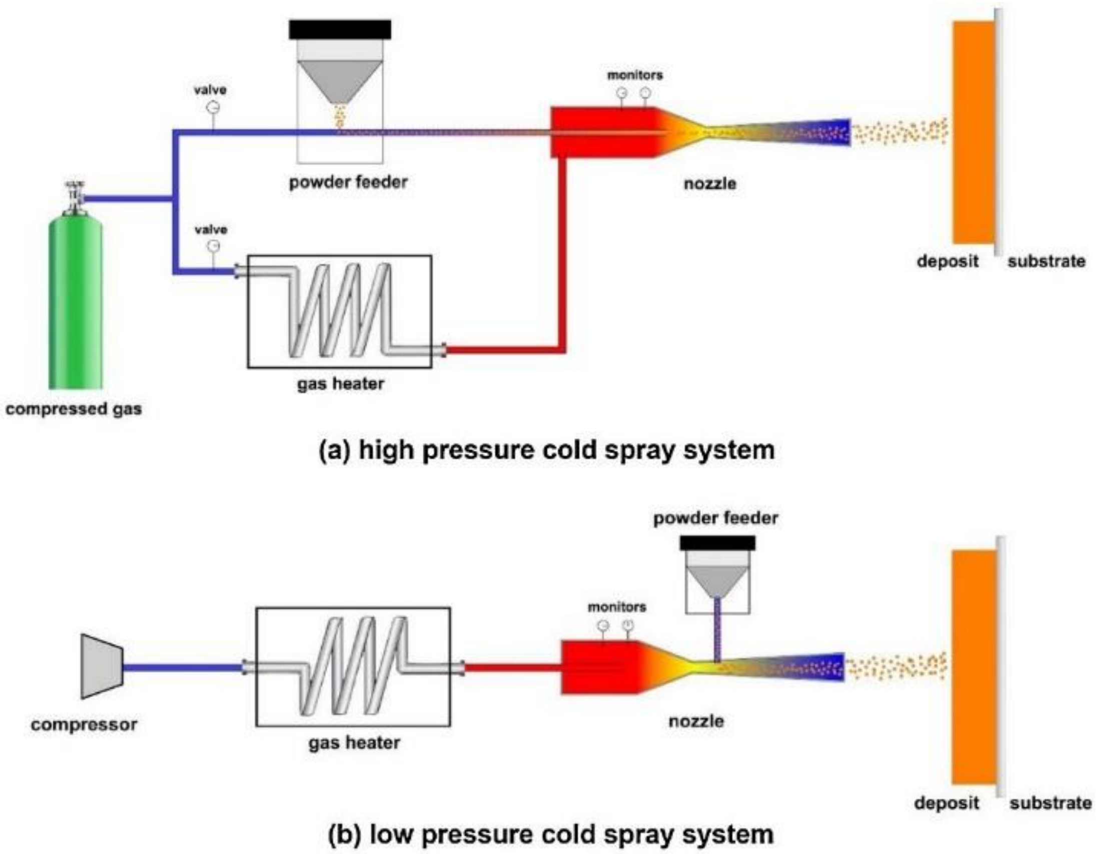


Figure 2: (a) High pressure, (b) Low pressure Cold Spray systems [5]

deposition [14]. If particles are smaller or greater than this recommended range, thus will lead to failures in deposition due to complications while accelerating [15]. Overall, in order to reduce this threshold of velocity, a larger particle and/or a greater particle temperature is desired as it is being deposited [16]. If this critical impact velocity is not reached the particles can bounce off or not adhere properly to the surface or substrate as shown in Fig. 3 [12].

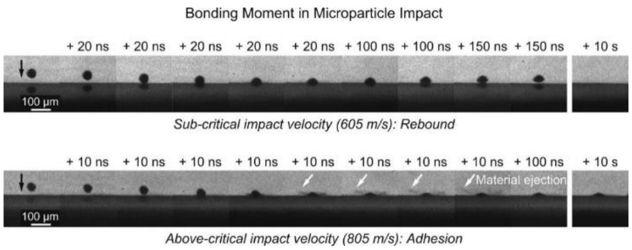


Figure 3: Impact Velocities [12]

3. COLD SPRAY ADDITIVE MANUFACTURING

3.1 CSAM Basics

Understanding the basic layouts, functions, and parameters of a Cold Spray system allowed it to be further developed for the Additive Manufacturing industry. Cold Spray Additive Manufacturing known as CSAM, is fairly new to the AM market. With over 30 papers written regarding this new method, starting in 2015 and peaking in 2019, this is a hot topic to a variety of industries worldwide. This AM technique uses the original Cold Spray process but in a modified and optimized way, in order to be used to manufacture parts. Just like Cold Spray, CSAM also uses fine metal powders that are accelerated in a system to supersonic velocities, in order to be imbedded onto a substrate [5]. These particles are imbedded in the same manner as traditional AM, layer-by-layer to form a free standing product. Unlike most AM processes, the materials used in CSAM do not undergo a thermal deformation, melting, or diffusion from its original state [6]. This enables a variety of desired features from parts created through CSAM.

The layout of a CSAM system is a compact condensed version of the entire Cold Spray machine. A 6-axis robotic arm is typically used to hold the whole unit in order to spray any desired structure needed, as shown in Fig. 4 [17]. Do to the fact that a free standing part is being created through this method, instead of a coating as Cold Spray was initially designed for, a variety of parameters specific to CSAM are needed. These parameters include key aspects of gas, powder, and nozzle optimizations to achieve desired outcomes [5].

3.2 CSAM Parameters

The gas used to accelerate the particles to the substrate is critical for deposition abilities. Ideally wanting a higher particle velocity requires the use of helium, but due to high production costs, nitrogen, air, or a mix of both is mostly used instead [7]. Helium is ideal due to its low molecular weight properties allowing particles to travel at a higher rate than nitrogen or air [15]. Higher particle speeds allow for precise deposition and improved properties of deposited materials [15]. Another way to increase velocity, is to increase gas pressures and temperatures [5]. Within a certain limit of materials, it is more effective to increase temperatures of the accelerating gas, than pressures of the entire system [15].

Figure 4: GE's CSAM Setup [17]

Since this system uses fine powder particles, it is very important that particle size, type, and amount is controlled [16]. More specifically, it is critical to maintain a constant and sufficient flow of material while operating, due to the vast number of outcomes that can occur. Overall, the powder feed rate can significantly affect the flow of powder passing through the nozzle, thus effecting the particle velocity impacting the substrate [18].

Other aspects that could be adjusted are nozzle parameters such as traverse speeds, scanning steps, standoff distances, spray angles, and trajectories of the overall systems [5]. All of these aspects can be depicted in a schematic shown in Fig. 5. Every time a CSAM system is used, an adjustment of all of the controls is necessary to achieve desired results. The development of generalized parameters is currently being developed for each application industrial partners are using this technology for [16].

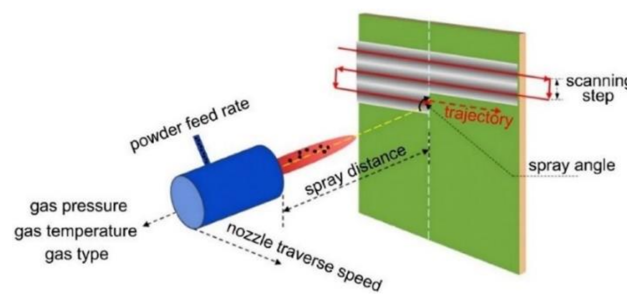


Figure 5: CSAM Schematic [5]

3.3 CSAM Properties

After fine tuning the parameters to be able to spray, an array of properties is analyzed to determine the quality and composition of the sprayed part. Key properties that are crucial to spray quality are the number of defects, porosity, microstructure composition, mechanical properties, bonding, and residual stresses within the part [5-8]. Defects can come in forms of micro-pores, micro-cracks, and vacancies within and on the surface of the part [5]. These occur when particles do not bond correctly, do not plastically deform fully, and/or are not deposited with the correct velocity, when sprayed [10].

All of these aspects can be determined by analyzing the part's microstructure composition. They all correlate with each other when it comes to determining the mechanical properties of the overall deposit [12]. Due to the nature of the process, it is typically known that without further treatment, the deposited structures exhibit very poor mechanical properties such as poor ductility and a weak structure [19]. Some of these poor properties are due to the buildup of residual stresses which is a containment of each layer's peening stresses within the parts being sprayed [19]. It has been proven for some materials, that a post heat treatment known as annealing is beneficial for the reduction of porosity and residual stresses, while increasing the bonding aspects and overall mechanical properties [20]. This can be clearly seen when looking at the differences in microstructures of freshly sprayed parts versus a post heat treated parts. As shown in Fig. 6, it is clearly noticeable that there is a difference in microstructure before and after heat treatment. After annealing, the particles recrystallize thus improving the quality of the structure by removing most defects [20]. Overall, these properties are being analyzed continuously by researchers for optimization in order to apply CSAM to a greater range of industrial applications. Additionally, the implementation of a laser-ultrasonic inspection device, to inspect created parts, helps further the quality that can be produced [21].

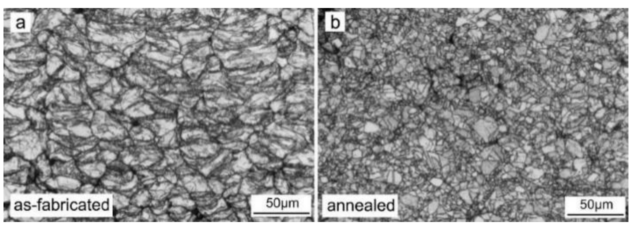


Figure 6: (a) Pre-treatment, (b) Post-treatment [5]

3.4 CSAM Machining

Heat treatment is not the only post processing method that can be done to a sprayed part. After a part is made, it is more than likely very rough in texture and porous [5]. Depending on the application, this might be an undesired characteristic, due to the fact that the part might not have all of the features, surface finishes, nor have the tolerances needed for its application. In order to achieve these desired aspects, machining is typically performed to create holes and to finish parts with certain dimensions [5]. Standard machining equipment can be used such as a Lathe, CNC machine, or a milling machine. An example of a CSAM part being machined to final specifications, is the Copper Flange shown in Fig. 7 and Fig. 8. These were created by an industrial company Plasma Giken, in Germany [22]. The first figure shows the part being sprayed, and also it being machined down to size on a Lathe. The second figure shows the addition of holes for attaching the flange, and the overall final surface finished part.

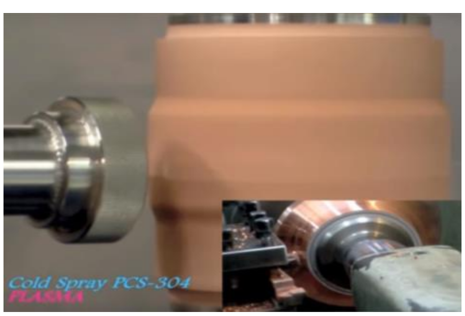


Figure 7: Copper Flange Sprayed & on Lathe [22]

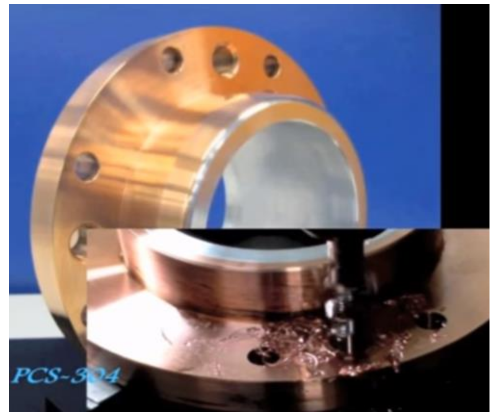


Figure 8: Copper Flange Drilled & Finished [22]

3.5 CSAM Robot Path

This leads to the discussion about robot paths, the effects of spray patterns on shapes and their properties, and the interlayer effect. These play a vital role for this process to perform in a uniform manner. The main component to this system other than entire Cold Spray setup, is the robotic arm that is controlled precisely; to be able to get desired results. This robotic arm in turn controls the nozzle of the whole system. Adjusting the nozzle parameters such as traverse speeds, scanning steps, standoff distances, spray angles, and trajectories, as depicted in figure 5, correlates to the overall robotic control [5]. In order to program the robotic arm to spray as desired, engineers use an off-line programming platform aided by a Computer-aided Design (CAD) / Computer-aided Manufacturing (CAM) software; similar to those used for CNC machines [5]. Off-line meaning computer developed control, instead of on-line which means person trained/guided control. These allow for the precise control of the robotic arm's path to the desired parameters. As shown in Fig. 9 (a), the robotic path is laid out layer-by-layer indicated by the green lines which indicates how the nozzle will be oriented. In part b, the final sprayed part is shown.

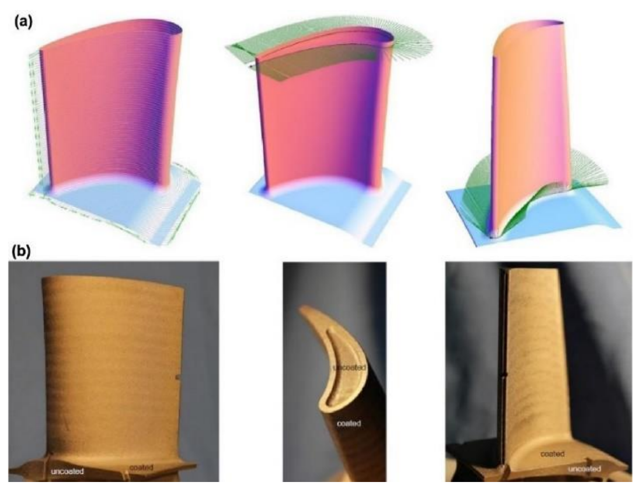


Figure 9: (a) Nozzle trajectory (b) Sprayed final part [5]

3.6 CSAM Robot Control

Not only parameters, but this control leads to the spray patterns that inversely affect the outcome of sprayed shapes and their individual properties. Out of the parameters of the nozzle, that is required to be adjusted and fine-tuned, the adjustment of its traverse speed which is controlled by the robotic arm is shown in Fig. 10. Depending on the robotic arm's traverse speed a unique pattern/buildup of material occurs as deposition happens. This in turn leads to a variety of resulting properties of the parts. As the speed of the nozzle increases a thin deposition is obtained, however, porosity increases due to inability for the particles to adhere to the surface properly [5]. On the contrary, as the traverse speed decreases, the density, residual stress, and thermal stress increases [5]. With thicker dense layers, there is a reduction in mechanical properties of the deposit, such as the elastic modulus, deposit strength, and adhesion strength [5]. Having less adhesion, leads to the delamination of layers caused by high residual stresses [19]. Since low speeds allow for a greater deposition rate, this in turn causes thermal stresses at the interface of the particles within the part [5]. These spray patterns can also be seen in figure 9 (b) as a striped pattern is evident on the surface of the part.

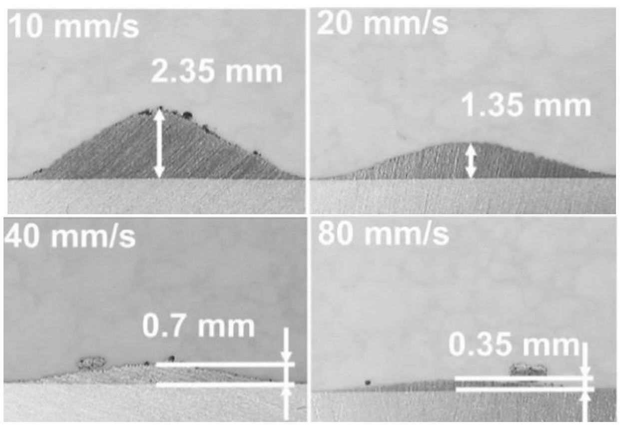


Figure 10: Nozzle speed effects [5]

3.7 CSAM Interlayer Effect

The interlayer effect within CSAM occurs between the particle and the substrate, as well as between two particle layers when deposited [5]. As mentioned before, due to the supersonic velocities these particles are sprayed at, they experience plastic deformation to allow them to mechanically interlock with each other [7-8]. The initial layer that attaches on to the substrate is important to ensure subsequent layers are deposited uniformly. Within these interlayers, if voids or particles that do not reach the critical velocity gets trapped, this will lead to an uneven, porous layer [5]. This in turn will force other interlayers between particles to also exhibit voids and potentially not mechanically interlock properly [7]. Overall, it would result in weaker mechanical properties compared to uniform dense layers [8].

3.8 CSAM Applications

CSAM is currently used for a variety of applications and is being researched to expand this area greatly. Applications as expressed in figures 7 and 8, show simple metal pieces being created, as a very common use of this technique. Researchers have recently written papers and are fine tuning ways to use CSAM for metal matrix composites, creation of magnets, making of heat sink fins, and creating complex structures [23-26]. Creating matrix composites using nano particles is a developing topic in material science, in which novel powders are evaluated to be sprayed with CSAM [23]. Another complex application entails a novel technique in which two 6 axis robots are used to obtain a helical shape within a hollow cylinder, as shown in Fig. 11 [24]. This effective approach was implemented in electric motor fabrication without additional assembly steps [24]. Further examples of the abilities CSAM can accomplish are expressed in Fig. 12 and Fig. 13. Figure 12 illustrates GE's demonstration to their CSAM machine generating a helical gear on a cylindrical rod [25]. Finally, figure 13 expresses ongoing research in creating optimal heat sink fins for efficient heat transfer purposes [26].

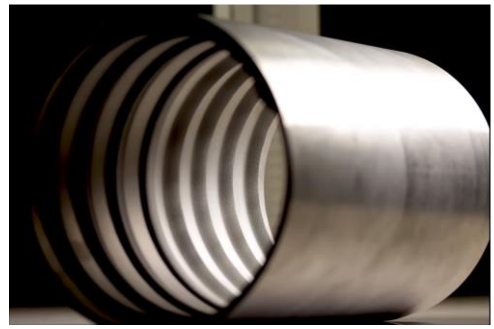


Figure 11: Magnet sprayed inner helical shape [24]

Figure 12: GE Spraying on Cylindrical Rod [25]

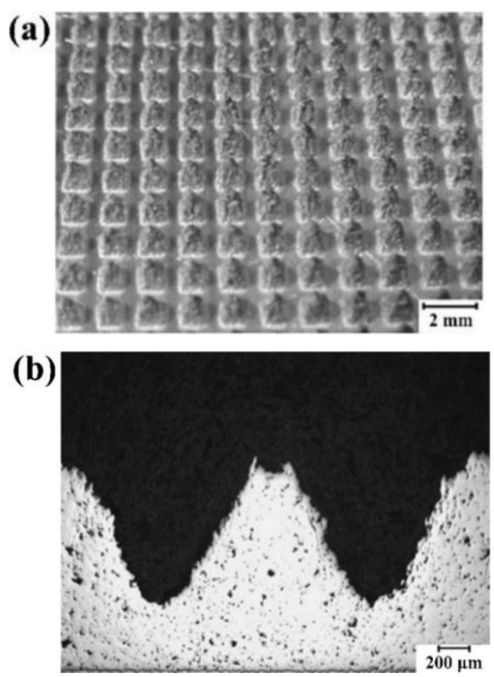


Figure 13: Heat Sink Fin (a) Design & (b) Cross-section [26]

In addition to complex applications, another basic need of CSAM is due to the fact that it can be used on surfaces that are sensitive to higher temperatures; such as nano-crystalline and amorphous materials [27]. These materials are extremely delicate and would not survive a standard thermal deposition [27]. Other applications that CSAM is used for is the repairing of damaged parts, welding parts together, and making additional modifications to existing pieces [28-29]. Major manufacturers, the automotive industry, and the military are substantial users of this technique. Their main use is to repair damaged structures such as crucial fighter jet components or basic vehicle frames, as shown in Fig. 14 [28]. CSAM has also been researched and developed to be an alternate method for welding parts together [29].

Figure 14: Repaired Damage Frame [28]

3.9 CSAM New Research

Newest research aims to help aid the full understanding of this process in ways, such as three-dimensional modeling and the creation of Neural Networks [30-31]. Being able to 3-D model and simulate allows for the understanding of limitations and challenges associated with CSAM [30]. This can overall aid to improvement of the quality and precision of particle deposition [30]. Neural Networks were developed to determining profile predictions, which may be used to improve geometric control with appropriate process planning algorithms [31].

Not only in a Neural Network analysis, robot control and path mapping are truly the key aspects to CSAM in order to create anything desired. This entails the use of complex programming and deep learning for the robotic components [31]. Shown in figure 7, a CSAM machine is spraying copper particles on to a vertical rotating mandrel, expressing one way of deposition [22]. An example of controlled movement planning is shown in figure 12. The use of a 6 axis robotic arm and a rotational turn table is implemented to systematically build a helical gear on the tip of a cylindrical rod [25]. Finally, a very complex example of a precisely controlled use of two 6 axis robotic arms is shown in Fig. 15. With the aid of artificial intelligence and machine learning, these robots have the ability to expand the endless possibilities this process can encompass [32]. All of these examples express a controlled robotic system in which a vast number of parameters, as previously discussed, are controlled and adjusted for a particular application.

4. CONCLUSION

Additive Manufacturing has come a long way since its 36 year presence. The process of making parts using a layer-by-layer approach has been redefined to an optimal and innovative way of manufacturing parts. Cold Spray Additive Manufacturing is one of the newest process researchers and industries worldwide are using

Figure 15: Two GE Robots working together to print one piece [32]

fine powder particles to create complex components. As discussed throughout this paper, the fundamental growth, process, and parameters, were reviewed to provide an understanding of this technology. This included an analysis of the Cold Spray system, the pre and post processing breakdown in CSAM, and the newest research areas this process has been implemented in. It is believed that this technology will expand greatly within the next five to ten years for researchers worldwide. Not only will it expand but it will be a replacement manufacturing method for a great amount of manufactured goods in the industry. Overall, Cold Spray Additive Manufacturing is an emerging technology that will provide beneficial outcomes in research and applications for a majority of industries.

5. ACKNOWLEDGMENTS

The author would like to acknowledge Dr. Arvind Agarwal, Dr. Cheng Zhang, Dr. Ibrahim Tansel, and the Mechanical and Materials Engineering Department at Florida International University, for their continuous support.

6. REFERENCES

[1] US. Patent 4,575,330, Charles W. Hull, Apparatus for Production of Three-Dimensional Objects by Stereolithography, 1984.

[2] Varotsis, Alkaios. "Introduction to FDM 3D Printing." 3D Hubs

[3] "Metal X 3D Printer from Markforged - Metal Printing Made Affordable." Mark3D UK - Markforged Partner for 3D Printers in the UK, 20 Sept. 2019.

[4] A.P. Alkhimov, V.F. Kosarev, N.I. Nesterovich, A.N. Papyrin, 1980. Method of applying coatings (SU 1618778).

[5] Yin, Shuo, et al. "Cold Spray Additive Manufacturing and Repair: Fundamentals and Applications." Additive Manufacturing, vol. 21, 2018, pp. 628–650., doi:10.1016/j.addma.2018.04.017.

[6] Li, Wenya, et al. "Solid-State Additive Manufacturing and Repairing by Cold Spraying: A Review." Journal of Materials Science & Technology, vol. 34, no. 3, 2018, pp. 440–457., doi:10.1016/j.jmst.2017.09.015.

[7] H. Assadi, H. Kreye, F. Gärtner, T. Klassen, Cold spraying — A materials perspective, Acta Mater. 116 (2016) 382–407

[8] M.R. Rokni, S.R. Nutt, C.A. Widener, V.K. Champagne, R.H. Hrabe, Review of relationship between particle deformation, coating microstructure, and properties in high-pressure cold spray, J. Therm. Spray. Technol. (2017) 1–48

[9] R.N. Raoelison, Y. Xie, T. Sapanathan, M.P. Planche, R. Kromer, S. Costil, et al., Cold gas dynamic spray technology: a comprehensive review of processing conditions for various technological developments till to date, Addit. Manuf. 19 (2018) 134–159

[10] F.Meng, S.Yue, J.Song, Quantitative prediction of critical velocity and deposition efficiency in cold-spray: a finite-element study, Scr. Mater. 107 (2015) 83–87

[11] F.F. Raletz, M. Vardelle, G. Ezo'o, Critical particle velocity under cold spray conditions, Surf. Coatings Technol. 201 (2006) 1942–1947

[12] M. Hassani-Gangaraj, D. Veysset, K.A. Nelson, C. Schuh, In-situ observations of single micro-particle impact bonding, Scr. Mater. (2017) 9–13

[13] "Cold Spraying - Cold Spray Coating Deposition." TWI.

[14] M.R. Rokni, S.R. Nutt, C.A. Widener, V.K. Champagne, R.H. Hrabe, Review of relationship between particle deformation, coating microstructure, and properties in high-pressure cold spray, J. Therm. Spray. Technol. (2017) 1–48

[15] S. Yin, M. Meyer, W. Li, H. Liao, R. Lupoi, Gas flow, particle acceleration, and heat transfer in cold spray: a review, J. Therm. Spray Technol. 25 (2016) 1–23

[16] T. Schmidt, F. Gärtner, H. Assadi, H. Kreye, Development of a generalized parameter window for cold spray deposition, Acta Mater. 54 (2006) 729–742

[17] Newman, John. "GE Announces Use of Cold Spray for 3D Printing." Digital Engineering, 19 Nov. 2013

[18] O.C. Ozdemir, C.A. Widener, M.J. Carter, K.W. Johnson, Predicting the effects of powder feeding rates on particle impact conditions and cold spray deposited coatings, J. Therm. Spray Technol. (2017)

[19] V. Luzin, K. Spencer, M.X. Zhang, Residual stress and thermo-mechanical properties of cold spray metal coatings, Acta Mater. 59 (2011) 1259–1270

[20] P. Coddet, C. Verdy, C. Coddet, F. Lecouturier, F. Debray, Comparison of the properties of cold-sprayed Cu-0.5Cr-0.05Zr alloys after various heat treatments versus forged and vacuum plasma-sprayed alloys, J. Therm. Spray Technol. 23 (2014) 486–491

[21] Lévesque, Daniel, et al. "Laser-Ultrasonic Inspection of Cold Spray Additive Manufacturing Components." 2019, doi:10.1063/1.5099730.

[22] Giken, Plasma. "Cold Spray Manufacturing of Copper Flange." Plasma Giken Youtube, 2011.

[23] Xie, Xinliang, et al. "Cold Spray Additive Manufacturing of Metal Matrix Composites (MMCs) Using a Novel Nano-TiB2-Reinforced 7075Al Powder." Journal of Alloys and Compounds, vol. 819, 2020, p. 152962., doi:10.1016/j.jallcom.2019.152962.

[24] Lamarre, Jean-Michel, and Fabrice Bernier. "Permanent Magnets Produced by Cold Spray Additive Manufacturing for Electric Engines." Journal of Thermal Spray Technology, vol. 28, no. 7, 2019, pp. 1709–1717., doi:10.1007/s11666-019-00917-6.

[25] GE, Research. "GE Cold Spray Technology." GE Research Youtube, 2013.

[26] Cormier, Yannick, et al. "Net Shape Fins for Compact Heat Exchanger Produced by Cold Spray." Journal of Thermal Spray Technology, vol. 22, no. 7, 2013, pp. 1210–1221., doi:10.1007/s11666-013-9968-x.

[27] Ajdelsztajn L, Jodoin B, Kim GE, et al. Cold spray deposition of nanocrystalline aluminum alloys. Metall Mater Trans A. 2005;36(3):657–666.

[28] J.C. Lee, H.J. Kang, W.S. Chu, S.H. Ahn, Repair of damaged mold surface by coldspray method, CIRP Ann.- Manuf. Technol. 56 (2007) 577–580

[29] Li, Wenya, et al. "'Cold Spray +' as a New Hybrid Additive Manufacturing Technology: a Literature Review." Science and Technology of Welding and Joining, vol. 24, no. 5, 2019, pp. 420–445., doi:10.1080/13621718.2019.1603851.

[30] Garmeh, S., et al. "Three-Dimensional Modeling of Cold Spray for Additive Manufacturing." Journal of Thermal Spray Technology, vol. 29, no. 1-2, 2019, pp. 38–50., doi:10.1007/s11666-019-00928-3.

[31] Ikeuchi, Daiki, et al. "Neural Network Modelling of Track Profile in Cold Spray Additive Manufacturing." Materials, vol. 12, no. 17, 2019, p. 2827., doi:10.3390/ma12172827.

[32] Alhart, Todd. "Brothers in Arms: These Robots Put A New Twist On 3D Printing." GE Reports, 15 Dec. 2017

A Review of the CSAM Submission to FCRAR 2020

Morris, Melissa
Embry-Riddle Aeronautical University
morrim44@erau.edu

Estimated Utility of:
"Cold Spray Additive Manufacturing - Process Review"
Utility to Community: \$1
Utility to Society: \$1

You do not use the FCRAR format. The template was only given in LATEX this year. If you do not have a LATEX editor, you can take the word document for a previous year and make sure to change the footer to be for the current conference. It is a double-column format. If you need your images larger, then you can include them all at the end using the full page. You can make them span two columns in the text as well, just make sure columns align OK.

Figure 1 is a bit hard to read. You might want to consider changing the format so that there is a space between the data between each year. Also, I'd suggest not including 2020 at all since the data is incomplete and doesn't really help your case. It took me a second to realize why there was an arrow there. It is best to stop at 2019, which is appropriate for a work published mid-2020.

If possible, come up with sub-headings for the CSAM section (Section 3). It may make it easier for readers to navigate that section. You do have some minor grammar issues here and there including plural/singular (I can't find it now, so maybe it was just me) and capitalization (e.g. "lathe" instead of "Lathe" on page 6). I think there was one or two other small things. Not a huge deal, but it'll clean it up if you do a quick read-through to correct them. "you" is used in the first new paragraph on page 6, which is not the correct voice. I think that was the only place it appeared.

This is a review, possibly for a thesis. If so, perhaps making a note in the conclusion about where the work is headed? "Future Work." (If not, then it is fine as it is. It may just be personal curiosity.) Alternatively, a discussion section with "the author's thoughts" could be added if you had ideas about what was next, even if you were not going to do that. Again, not mandatory, it would just add value to the paper.

Rebuttal Reply for the Paper "Cold Spray Additive Manufacturing - Process Review"

Shah, Kumar
Florida International University
kushah@fiu.edu

Answer To Review:
"A Review of the CSAM Submission to FCRAR 2020"
Utility Review: $1

Trained Probabilistic Models for the NAO Robot in a Labyrinth

Samantha Eaton, Kleanthis Zisis Tegos, Sagar Vilas Jagtap, Akram Alghanmi, Godwyll Aikins, Neda Keivandarian, Ran Bi, Jesse Torres, Kholud Alghamdi, Marius Silaghi

Florida Institute of Technology

ABSTRACT

A framework based on a Probabilistic Model for the moving behavior of the NAO humanoid robot in the environment given by $20in \times 20in$ vinyl maze cells is being trained and made available for future applications. NAO is one of the most advanced humanoid robots, having advanced speech, vision, and behavior based on artificial intelligence already implemented on it, and being a precursory of the larger Pepper robot famous for being used as host at certain hotels. Pepper uses wheels, probably since the leg-based movement of Nao proved hard to harness with precision and robustness. Indeed, most of the intelligence currently present in NAO is speech and gesture-related, while its autonomous walking capabilities are only little exploited in existing available software, and only with reflexes without high-level utility-driven intelligence.

We test that it is possible to exploit a public NAO sensor database made recently available, to build a sample probabilistic model for walking and turning in a controlled vinyl maze. The probabilistic model is a new and powerful representation of related phenomena based on random variables and with conditional probability tables for the NAO sensors computed from experimental measurements given relevant environment states. The model allows for complex planners and reasoners, that are based on rich POMDP models, to be built on top of it.

Such a high-level AI framework allows for easily giving NAO new tasks by just specifying the corresponding utilities. The proposed model is tested with a simple particle filter localizer on a predefined trajectory, and improvements and data missing in the Nao database are being identified for future work.
Keywords: Humanoid, framework, model, AI, Nao.

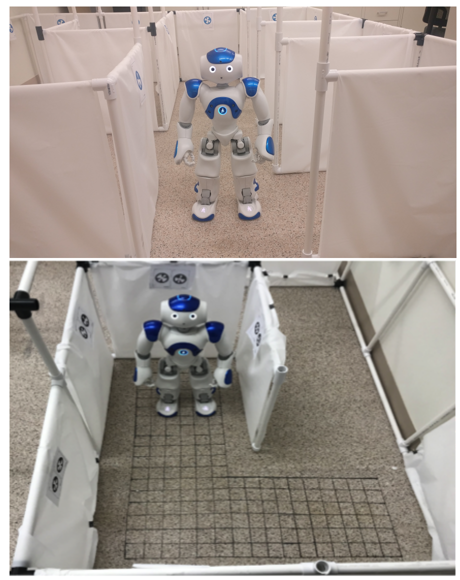


Figure 1. (a) Restricted Environment; (b) Markers for Standardizing Data Measurement

1. Introduction

The humanoid NAO robot, together with its wheeled larger version Pepper, are among the most advanced robots available to the general public. These robots are known to gracefully dance, walk, talk and entertain the users with intelligent dialog. However, the walking intelligence of these robots is rather low, and no common application enables NAO to robustly walk with high-level, (i.e., utility-driven) intelligence; not even in restricted environments.

Nao is being used for a robotic soccer contest called Robocup where teams of robots play without central controllers. However, the robots in such games are endowed only with reactive behavior for their role in the game and are not utility driven with high-level intelligence. Futher, the Robocup robots are based on a secret operating system not available to the rest of the public owning Nao robots, like us. Also, the games are organized on hard terrain that is optimal for Nao movements, but walking on other surfaces, like carpets, is much harder for it.

In previous work, groups of researchers in our lab started to address this problem by standardizing a restricted maze environment based on $20in \times 20in$ white vynil cells [1], Figure 1.a,b, and started organizing a public database publishing sets of contributed sensor data and movement measurements [2], usable for characterizing and training models of the interaction between Nao and such restricted environments. However, the test of the Nao sensor database was very preliminary and in this work we undertake the extension of the data gathering and framework formalization, as well as testing, for the Nao robot.

We now propose a general probabilistic model of the movement capabilities of Nao in a controlled environment, defined by the labyrinth of 20 inch square rooms with vinyl white walls.

Motivation. Our work aims to build a probabilistic model of the NAO robot's sensors and actions non-determinism, enabling the application of high-level intelligent algorithms for tasks in a vynil labyrinth. The tasks context include school maze problem solving contests for educating students in advanced approaches to artificial intelligence concepts like uncertainty and non-determinism.

The foreseen contest application is the navigation of the maze while making educated movement decisions based on the information available. Tasks can consist of guiding users in the labyrinth, patrolling, searching, transporting. With high-level intelligence, new tasks should be achievable simply by having users communicate to NAO utilities in terms of rewards for desired states. Decision components can include operations such as localization, mapping, and planning.

Contribution. In this particular work we identify and propose a particular Partially Observable Markov Decision Process (POMDP) model that is elegant, almost textbook level, and still represents sufficiently well the NAO robots' interactions with a vynil labyrinth to accomplish many tasks.

Further, we present a Dynamic Decision Network (DDN) representation of this POMDP, which allows us to incrementally and educationally show the construction of our NAO POMDP from components. This part is intended as a crash tutorial on POMDP/DDNs to the FCRAR community. As such the DDN is :

- firstly showed constructed as a Bayesian Network (BN) for static inferences of location based on the NAO's available sensor inputs (sonars and landmark detection),
- further enhanced to a Dynamic Bayesian Network (DBN) by including movement and transition models, and
- finally promoted to a POMDP/DDN by enabling the addition of rewards that specify tasks to do, in a utility-driven paradigm.

Lastly, we report on the experimental evaluations that prove and quantify the non-determinism and uncertainty of the NAO architecture, not only confirming the needs for frameworks like ours, but also actually providing the numerical Conditional Probability Tables (CPTs) components of the POMDP models which can be used for actual inference with the proposed framework. The CPTs are trained on a combination of new measurements and existing data in the NAO database [2].

Moreover, movement of NAO in the labyrith with our probabilistic model-based localization is executed based on the CPTs we construct according to the described mechanisms, illustrating applications and evaluating the accuracy achievable using the BN/DBN parts of the proposed POMDP model.

2. Background and Related Work

Let us introduce the main background on the Nao robot, on dynamic Bayesian networks and on Partially Observable Markov Decision Problems (POMDPs), which are among the most principled general frameworks in artificial intelligence for addressing robotic problems.

2.1 The Nao Humanoid

Nao is a humanoid [3], that has advanced capabilities being able to talk and walk. His sensors consist of two sonars placed symmetrically on the right and left of his torso, as well as two cameras placed one above the other between the LED eyes. The cameras come with software to detect a special type of landmarks, called Naomarks.

Among the closest related humanoid robots that are available we mention SoftBank's Pepper, that has wheels locomotion instead of legs, and Honda's Asimo [4] who also has a camera on his face but two asymmetrical sonars on its torso, of which one is specialized for detecting the ground. Asimo has walking that is more advanced, being able to climb stairs, but it is not available for sale.

2.2 Relevant Nao Sensors and Actuators

The only relevant sensors in our approach are the two Sonars and the Landmark detector. The Landmark detector is a high-level sensor implemented in software on top of the Nao cameras. While this detector works relatively well in static scenarios, the version in the standard edition we have fails frequently at detection when the robot is executing walking tasks. These failures have to be modeled and accounted for by the high level intelligent architecture and this sensor non-determinism is also quantified in this work.

Similarly the only actuator exploited is the `walk` command, whose analysis and probabilistic model is the subject of the second part of Section 4.1. While Nao has multiple walk commands, for now we restrict ourselves to three instances:

- step 2 inches forward
- turn 10 degrees right
- turn 10 degrees left

Rotations of 5 degrees are also studied but their non-determinism was observed to be significantly higher in preliminary evaluations, as frequently the robot just does not achieve any displacement on receiving the command.

2.3 Bayesian Networks

A Bayesian network is a graphical probabilistic model representing conditional independence information with graphs consisting of nodes and edges. A node represents a random variable, and an edge represents a direct conditional dependency between the random variables (nodes) it links. Edges are directed with parents preferably being selected as causes of descendants, which simplifies estimation of conditional probabilities. Nodes are conditionally independent of indirect ancestors in the graph given certain nodes in-between (a Markov blanket). The domains of the random variables could be either discrete or continuous. The random variables in the Bayesian networks can be classified in three types: observable (evidence), non-observable (hidden), and controllable (also evidence, graph source nodes). Generally Bayesian network graphs are constructed to be acyclic [5]. Every node is associated with a conditional probability table specifying its conditional probability given all possible value combinations for its parent variables in the graph, or some compressed representation thereof.

A Bayesian Network is a compressed representation of the full joint probability distribution of its variables. The main application of using Bayesian networks is the enabling of probabilistic inferences in environments where state or phenomena are uncertain or non-deterministic.

A Dynamic Bayesian Network (DBN) is a Bayesian network of a special type, where sets of nodes are repetitions of a pattern in a sequence that can potentially be infinite. It is frequently used to represent sequential processes, each repetition corresponding to a time step, where the time is discretized. It thus represents a probabilistic temporal model. Dynamic Bayesian networks are generative models. The graph in DBN is composed of a sequence of time steps or slices, namely $(0,\dots,t-1,t,t+1,\dots)$ [6].

As an example, Particle Filtering, one of the main reasoning techniques for filtering in DBNs, works as follows. After

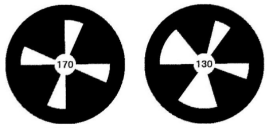


Figure 2. Standardized wall markers, in pairs

(i) an initial population at time t with N samples from the hidden state variables set x_t is created from the distribution $P(X_t)$,
(ii) firstly, for the subsequent time step $t+1$, its next state value x_{t+1} sample is generated based on the transition model $P(X_{t+1}|x_t)$
(iii) and then weighted by the likelihood that it assigns to the corresponding new evidence e_{t+1}, $P(e_{t+1}|x_{t+1})$.
(iv) Lastly, this weighted population is interpreted as a probability distribution $P(X_{t+1})$, that in a subsequent round is resampled per the aforementioned procedure to generate the next set of samples, thus repeating these inference operations at each remaining time step.

2.4 POMDPs

The Partially Observable Markov Decision Process (POMDP) is a framework for modeling dynamic uncertain environments, adding rewards associated to states and/or transitions in Dynamic Bayesian Networks. A POMDP is defined by a set of states, a set of possible observations, a set of actions, a transition probability function between state repetitions, observation conditional probabilities, an initial belief, a discount factor, and rewards associated to states.

Through observation of evidence, the robot will update the probability distribution of its current state (aka. belief) in order to optimize its choice of the next action maximizing future rewards. Actions at a time step contribute in the next state transition.

A reward represents the benefit of a state (and potentially, action). The typical goal of reasoning in a POMDP is to generate an optimal policy (map belief-action) which maximizes the amount of future rewards [7]. A belief is a probability distribution over states. The robot will optimize the next action, making decisions, based on its current belief [8].

One way to depict POMDPs graphically is based on extensions of DBNs called Dynamic Decision Networks (DDNs). In DDNs, each time step is associated with the set of actions that can be performed in it, treated as additional evidence factors, as well as with a reward that depends of state and actions.

2.5 The Nao database structure

We use the database of experimental measurements constructed by the previous work in [2]. This database contains sensor and transition measurement data, as well as its documentation and relevant code snippets. The sensor data contains the measurements obtained from the various sensors on the NAO robot; the left and right sonar sensors for detecting objects and the visual sensor for detecting the standardized landmarks on the walls, see Figure 2. The transition data is measured as the center of mass displacement when the robot is given the command to move between two adjacent squares, each $2in \times 2in$, in the labyrinth.

The transition data is organized into 4 column datasheets. Each datasheet describes experiments with a given action, and its columns are: the given position, given orientation, recorded position, and recorded orientation. As the name of the columns suggests, each datasheet records actual responses measured for the command associated with it.

The sensor data consists of 12 column datasheets (see Figure 3).

- Two columns were dedicated for the position of the robot in the cell of the labyrinth, namely the "Row" and "Column" in the maze.

- "Angle" is the degree of the rotation of robot with respect to the reference vector.

- "Head Pitch" and "Head Yaw" (the orientation of the robot's head) are represented in the sensor datasheets but were controlled at the still position and not used in the localization of the robot.

- The "Left" and "Right" sonar sensor readings of the robot were the most robust Sensor data readings.

- In the Nao sensors database data, two landmarks are placed on each maze wall center, and data pertaining to their detection was recorded in the sensor data: "Landmark detection" (whether the landmark was detected), "Landmark ID" (Number ID on the Nao Mark), "Alpha" and "Beta" (which are the line of sight angle of the landmark's position from the robot's head axis), and "Width" and "Height" of the detected landmark.

2.6 Related Work

In [9], a Nao robot is addressing the problem of simultaneous localization and mapping (SLAM) in a room which has multiple Nao marks with random locations. The contribution proposed to enhance SLAM addressing the real-time incorporation of new landmarks in exploration.

An autonomous algorithm to obtain the parameters of the POMDP in a navigation system for indoor environment based on WiFi and ultrasound observations is implemented in [10]. This algorithm can learn the observation and transition matrix in autonomous mode which is coming from wifi simultaneous localization and mapping so it could obtain the WiFi and Ultrasound environment map with minimum effort. The localization algorithm converges faster by adding a global navigation system sensor.

A multimodal interaction system between a human and a robot was suggested in [11] by expanding a POMDP framework in a humanoid robot called "Pepper" to observe various multimodal information streams with its own sensors. The interaction system is made of two strata in the framework of POMDP – set the first stratum so as it decides to start interaction based on the physical distance between Pepper and a user, and set the second stratum to control multimodal interaction. The effort led to stratified interaction to reduce the increase of the user's belief states, making the problem more manageable. It shows that POMDPs were previously used for emotive behaviors, even if not for walking, with such robots.

A variable resolution technique is proposed in [12] for reducing the complexity/state space of the POMDP. This is accomplished by automatically adjusting the number of the states in the grid based on features of the environment map while maintaining the level of detail required for planning at a given region to find a robust and efficient policy. This is validated in a POMDP-based simulation of a navigation environment, a realistic robot simulator, and an autonomous robot.

col, row	Represent the location variable in DBN; has discrete values [(0,0),,,, (3,7)]
angle	correspond to orientation of robot, represent angle variable in DBN, has discrete values [0, 10, 20, ...,320]
head pitch, head yaw	Represent the orientation of the head
sonar left, sonar right	Sensor reading for distance to the wall
Landmark detected	Represent Boolean values; True, False
Landmark id	Represent the type of id, there are three types.
Alpha, beta, width, height	Properties of a landmark, direction of view and dimension

Col	Row	angle	Head pitch	Head yaw	Sonar right	Sonar left	Landmark detection	Landmark id	Alpha	Beta	Width	height
0	1	10	NA	NA	.45	.47	True	68	-0.00892	-0.0649	0.09798	0.09798
3	5	230	NA	NA	.48	.26	True	85	0.219224	-0.1276	0.18434	0.18434

Figure 3. Sensor Inputs, as organized in the Nao Database

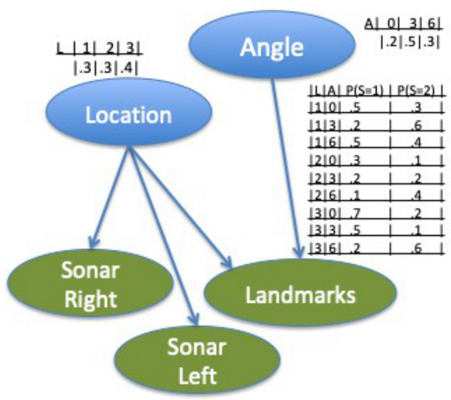


Figure 4. Conceptual Belief Network for Observing the Situation

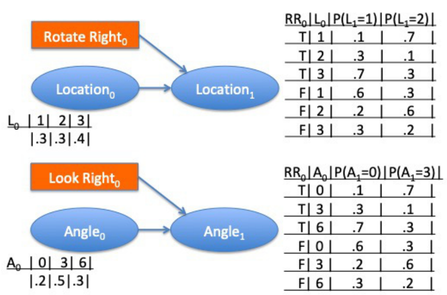


Figure 5. Conceptual Belief Network for Transitions

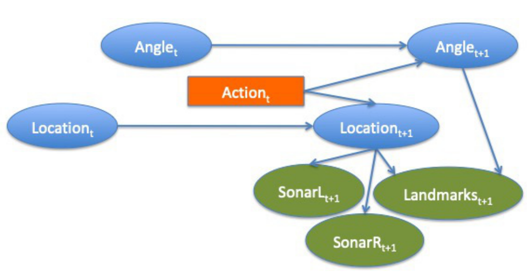


Figure 6. Dynamic Belief Network utilized

3. Concepts and Data Structures

To model the phenomena taking place when a Nao robot is observing or walking in the maze, belief networks are designed to generate the CPTs from sample data, as conceptually shown in Figures 4 and 5.

The Bayesian Network in Figure 4 abstracts the relations between: the relative position or "Location" of the center of the robot (the center of the space between its feet) with respect to walls, the torso heading angle, the angle between its head and torso, the measurements of its sonars, and the landmark detection (the actual sets of values of the variables are more complex, and described later).

The Bayesian Network in Figure 5 describes abstractly the impact of actions on location and angles, for rotations. Similarly, the actual sets of values of the variables are more complex, and described later. With most types of walking, both the previous angle and previous location have an impact on the resulting location.

The Location or position node holds one of the M unique $x-y$ coordinate points within the labyrinth, and possible heading angles of the torso. If the maze would have m rows and n columns, then $M = m \times n$. The Angle node represents the angle between the head and torso, as controlled by the robot.

The Left Sonar node represents the measured value of the left sonar based on the $x-y$ coordinate point within the labyrinth and the overall angle the robot body is facing. Its conditional probability table (CPT) is a table of size $M \times N \times O$, where M represents the total number of unique $x-y$ coordinate points within the labyrinth, N is the number of unique torso heading angles, and O is the number of possible sonar readings. The Right Sonar node is defined similarly, but represents the measured right sonar values.

The landmark node represents the event of detecting some of the landmarks in the environment. It is related to the $x-y$ coordinate position within the labyrinth and the overall position angle of the head of the robot. Its conditional probability table (CPT) is compressed based on an independent success assumption into a table of size $M \times N \times L$, where M represents total number of unique $x-y$ coordinate points within the labyrinth, N is the number of possible unique sums between torso heading angle and the torso-head angle, and L is the number of possible landmarks.

A possible value for the Landmark random variable is any bag of landmarks with their possible sizes and positions in Nao's visual field. The representation can be compact by using a single Boolean per landmark to represent it's visibility, while the distribution of the rest of the landmark location and size values can be integrated as deterministic nodes with Gaussian noise.

This representation is more general and flexible than the earlier version [2] where there are three landmark nodes per time frame,

Table 1. Sonar Left Node

X	Y	Angle	P(Sonar=0.46)	P(Sonar=0.47)	...
0	0	0	0.25	0.000	...
0	0	11.25	0.000	0.000	...
...	...	...	...	...	...
m	n	N	...	...	...

Table 2. Landmark Node

X	Y	Angle	Landmark 1	Landmark 2	...
0	0	0	0.8	0	...
0	0	11.25	0.8	0	...
...	...	...	...	...	...
m	n	N	...	...	...

that can recognize up to maximum three landmarks, an assumption that is not always true with complex mazes. The bag representation also simplifies the handling of matching different landmark IDs in different locations. The case of landmarks with repeating IDs is described later.

While the shown figures display conditional probability tables for illustration, the actual conditional probability tables of these Bayesian networks are estimated by counting from the measurements in the Nao sensor database, as described further in the article.

4. Models and Algorithms

The database extension we provide offers algorithms for estimating the belief (i.e., conditional probability distribution), given current sensor measurements, as well as for filtering the new beliefs along sequences of actions and measurements of sensor readings .

4.1 Training Conditional Probability Tables

Building on top of the sensor data made publicly available by a previous work [2], libraries for estimating the conditional probability tables for our Belief Network (refer to Figure 4) were developed.

The proposed belief network is **discrete**. While location is a continuous quantity, the state space was divided into discrete cells. The same approach was used for the sensor measurements, which are also continuous, but were categorized into discrete groups.

The Sonar nodes each relate data from the respective sonar sensors with the location and angle information. Extracting the sonar data for each sonar results in a $(m \times n) \times N \times O$ matrix, where N is the number of torso heading angles recorded at each position and O the number of sonar readings handled. As some measurement readings never occurred in the training set leading to zeroes as shown below, the probability mass is later redistributed to compensate for possible noise by assigning a small probability to each possible value.

For the Landmark node, we separately represent the Boolean landmark detection event and its actual parameters in terms of position. When a landmark is detected its position is pretty accurate and therefore can be approximated with a deterministic function with small Gaussian noise. For the Landmark detection event, the CPT can be represented compactly by exploiting an *assumption of cause independence between individual landmarks' detection success*. As such one only needs to store separately the probability of detecting each individual landmark, and the probability of each bag of landmarks can be inferred from these assumption of independence with the product rule. The resulting table is a $(m \times n) \times N \times L$ matrix, where L is the number of landmarks in the environment. At training the landmark data is True if the landmark was encountered, and False otherwise.

For the case where landmarks with the same ID are reused in the environment, the compact representation needs to include one column for each count of detected markers with identical IDs, and the aforementioned deterministic probability node linking the position to the parameters of the detection are normalized over all such repeated occurrences.

Table 3. Translational Transition Probability for Location Node

$State_{cmd}$, $Orientation_{cmd}$	0, -90	0, 90	...	8, 0
0, -90	0.8	0	...	0
0, 90	0.1	0	...	0
...	...	...	...	...
8, 0	...	...	...	...

Transition probabilities for the NAO humanoid robot were calculated using the transition data that came with the dataset. Recorded readings and each action command sent to the robot were used to compute the probabilities by counting.

$$P(r|0,S) = \frac{\text{Number of times state r = 0, and orientation S}}{\text{Total times the command was executed}} \qquad (1)$$

From the Nao database, the following translational conditional transition probability table of size $(l \times o) \times (r \times u)$ was produced, where l is the relative state the robot was asked to move to, o is the relative orientation the robot was asked to be in, r is the possible state the robot may be in, and u is the possible orientation the robot may be in after execution of a command. Both $l, r \in S\{0,1,2,3,4,5,6,7,8\}$, in which each number represents either the robot's current or neighboring position.

Specificity to the NAO robots. What makes this implementation specific to NAO is the structure and the parameters that each node represents. The sonar sensors are placed symmetrically right and left on the NAO's chest and therefore, the sonar data will only depend on the location and orientation of the robot's torso. The landmark node utilizes visual data from the cameras placed one on top of the other on NAO's head. Because the parameters that the network makes use of are highly dependent on their sensor's physical placement on the robot's body, the model and its implementation that we present to describe the robot behavior is unique to the NAO robot.

If this network was to be generalized for use, for instance, on another robot with cameras that the landmark node draws data from, fixed to the robot's torso; it would completely invalidate the network as the landmark node would not depend anymore on the head angles. It is however possible to use this network on another robot that shares the same physical features and sensors as NAO or with appropriate adaptation for robots that share similar features.

Specificity to the used labyrinth. All the landmark measurements are based on the positions of NaoMarks in the labyrinth so, inherently, the conditional probability tables (CPTs) of the computed model are only relevant in the context of the given labyrinth.

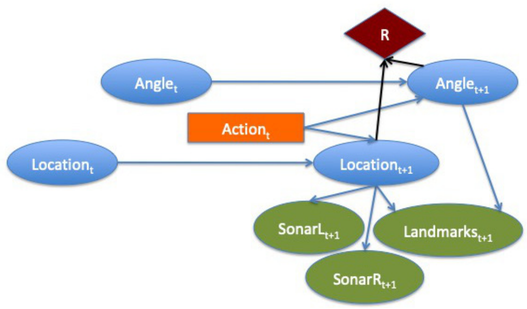


Figure 7. Dynamic Decision Network

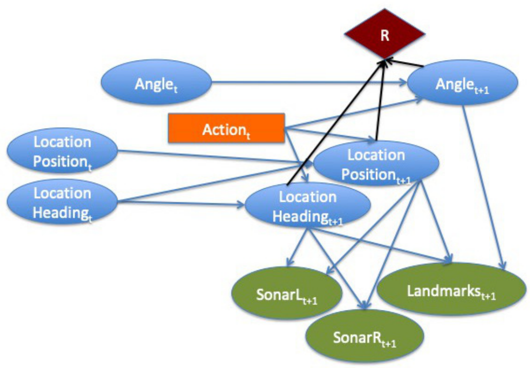


Figure 8. Dynamic Decision Network optimized

The readings of the sonar sensors also depend on the labyrinth wall material. Different materials reflect the sonar waves differently. Therefore its CPTs are also relevant only within this particular setting.

The labyrinth also provides the testing environment for the sample models test application described in experiments. Measurements of the sensors were extracted at various locations as described in the previous team's article [2], measurements which were used to train the models described in this paper. The labyrinth was used to measure the accuracy of the obtained models, in terms of number of times the network guessed the current location correctly, given its sensor measurements.

4.2 The POMDP Model Design

From the Bayesian Network concept describing observation and transition phenomena in Figures 4 and 5, we build a Dynamic Belief Network shown in Figure 6.

Performing exact inference for Dynamic Bayesian Networks (DBN) is hard for complex networks, given that the cost of the update procedure is O($d^{|x|+2}$) for domain sizes d with $|x|$ variables in a densely connected graph, as stressed in [5]. For stochastic methods sample generation is possible with various approaches. In the first approach, the distribution used in sampling comes from a pooled multivariate Gaussian mixture with weights given by the previous round samples. Practically, the distribution used for re-sampling is interpolated using a radial-basis function with Gaussian kernel. In the competing approach, the distribution used for sampling is obtained with multi-variate interpolation (Shepard method or linear) [13].

The Dynamic Decision Network. Integrating rewards conveying the utility of goal states, a dynamic decision network is obtained, as in Figure 7. The Dynamic Decision Network (DDN) is the graphical representation of POMDPs.

Since for many actions of the humanoid it can be assumed that the transition is independent for location heading and location position, the CPT storage space is improved with the finer grain dynamic decision network in Figure 8, where the "Location" random variable in the previous models is replaced by two random variables: "Location Position" and "Location Heading", for the $x-y$ coordinate and for the torso heading, respectively.

Rewards Specification. The rewards needs not be calculated by us, as it is supposed to be the way for end-users of our library to specify tasks. This is similar to how queries are not included in a first order logic knowledge database, but are posed by the end users. So, while we here provide an AI library of probabilistic models, the end-users of such models sets the rewards for achieving specific tasks.

For example, in some designs the reward for the POMDP could be proportional to the distance of a cell to any of the surrounding walls, as well as the expected distance to the end position. What this means is that the robot will try to choose the move that will bring the robot closer to the end goal, taking the walls into consideration.

5. Experiments

Preliminary experiments that enabled extensions to the Nao sensor database with additional landmark measurements were achieved.

The experiment consisted in measuring and evaluating sensor inputs for more situations, and analyzing the data into CPTs as reported in the article for better estimation of the scale of non-determinism. The CPTs, and related software for access, are too large for reproduction in the article but they are made available in the database.

5.1 Setup

As shown in Figure 1b, the humanoid was placed at the starting point with three immediate landmark pairs around it (left, right, and back). Markers were also drawn on the floor beneath the humanoid to standardize the observation of the change in position results.

A series of trials was executed using the humanoid to find the left and right sonar readings and the landmark visual readings based on the various $x-y$ coordinate points within the labyrinth and the torso and head positions. The generated data was saved in CSV format, added to the database, and used to generate the belief network.

5.2 Obtained Models

The belief network with sample node data is shown in Figure 4. The main inputs to the overall system is the $x-y$ position coordinate point and the angle of the head.

In our experiments, M is 32 for the unique $x-y$ coordinate points within the labyrinth and N is 32 for the unique degree angles (0 degrees to 360 degrees in steps of 11.25 degrees). Data in the database was available for $L=3$ distinct landmarks.

The Figures 9, 11, and 10 quantify the measured non-determinism in actions and sensors for Nao.

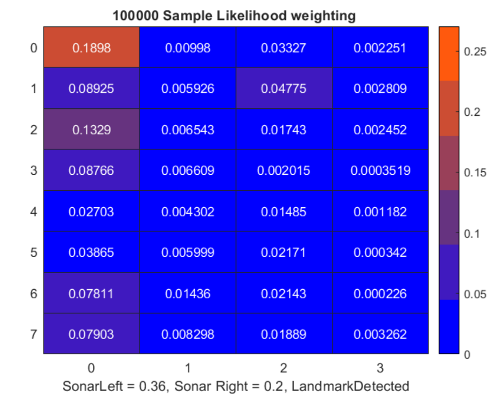


Figure 9. Sensor Uncertainty: The belief based on a sensors measurement, $P(S_L = 0.36, S_R = 0.2, L = T | X = x, Y = x)$

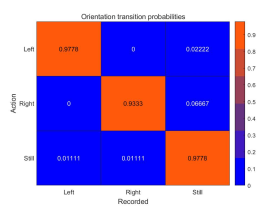


Figure 10. Nao 10° Rotation Non-determinism: The heatmap of beliefs concerning position after a 10 degrees rotation. The ordinate shows the command and abscisae shows probability of the obtained state

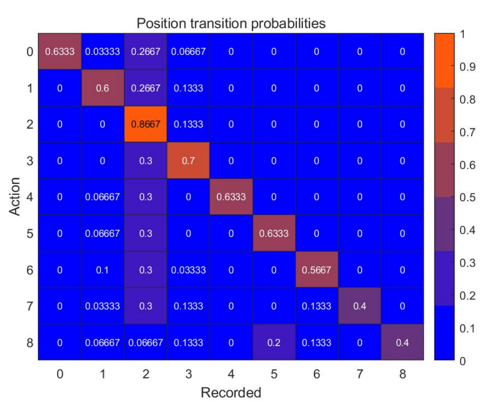


Figure 11. Nao Stepping Non-determinism: The heatmap of beliefs concerning position after a step

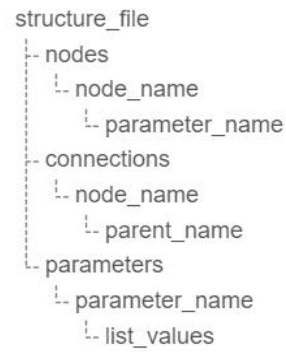


Figure 12. Layout of JSON file containing Bayesian Network

5.3 Model Guide

The goal of this work was to provide baseline models upon which the locational awareness of the Nao robot can be improved. This section serves as a reference on how the obtained models, which are made publicly available in the Nao database, can be used.

```
BN estimate_CPTs(BN_structure, training_data, BN)
```

The Python function `estimate_CPTs` calculates the conditional probability structures, given the Bayesian network structure and the corresponding training_data measurements, which are stored in a file. It accepts three file names, with the BN_structure and output BN files having JSON format, and the measurements having csv format. The format of the BN_structure is shown in Figure 12.

The requirement for the measurements file is that the first row contains the column names, which correspond to the nodes of the network. The output file only stores the matrices, and no structure information regarding the network, since that can be extracted from the structure file.

```
distrib belief(l_sonar, r_sonar, landmarks, theta)
```

The function `belief` is used to compute a probability distribution for all the possible current locations (x, y, α) within a cell given the sonar and landmark readings, as well as the head-torso angle `theta`.

```
distrib likelihood_weighting(DBN, query_vars,
                             evidence_vars, samples_nb)
```

The likelihood weighting function calculates the probability distribution over the query variables given the evidence variables. It accepts the DBN as a filename for conditional probability tables and for its structure, the query variables as a dictionary with keys being the needed node names, the evidence as a dictionary with keys being the node names and values being the measured node state, and a number of samples to be run. The more samples are run, the higher the accuracy of the likelihood weighting algorithm.

API of the Bayesian Network.
The Bayesian Network was developed using Python to parse the data results from the CSV files and generate the expected node results. The generated conditional probability tables (CPTs) are written to CSV files.

API of Likelihood Weighting Module.
The Likelihood Weighing Module was developed using Python. The code parsed the Bayesian Network conditional probability tables (CPTs), the user input specifying the positional goals of the robot the expected current position of the robot, and the robot's sonar and landmark detection data based on it's current position. The code produced the transitional probabilities for the next timestep at that location, and the estimated final robotic location with head orientation.

6. Conclusions

The problem of lack of support for easy development of intelligent and autonomous Nao robot applications with walking mobility is addressed by extending on the preliminary prior work in our organization, that introduced a "sensors and transition measurements database" for probabilistic reasoning in restricted environments.

This work extends prior contributions not only by adding new data measurements with new sensor and landmark sensor measurements, but also by contributing a general sensor model, and a Dynamic Decision Network framework with libraries for training conditional probability tables and for likelihood weighting inference steps compatible with the data formats in the Nao sensors and transition database.

References

[1] A. Hasanain, T. Weekes, M. Person, K. Paul, Y. Chang, A. Rothman, Z. Rui, R. Rahil, M. Syed, C. Wodd, and M. Silaghi, "Models and framework for supporting humanoid robot planning & exploration," in *FCRAR*, 2018.

[2] K. Alghamdi, J. Torres, J. Burden, V. Maddineni, R. Alharbi, T. Neelam, P. Tarsoly, A. AlHaif, J. Nke, S. Jadhav, S. Dodle, and M. Silaghi, "Towards sensor and motion measurements databases for training models for the nao robot," in *FCRAR*, 2019.

[3] SoftBank. https://www.softbankrobotics.com/emea/en/index, 2020.

[4] Honda, "Asimo technical information," 2007.

[5] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[6] K. P. Murphy, *Dynamic Bayesian Networks: representation, inference and learning*. PhD thesis, University of California, Berkeley, 2002.

[7] J. Luo, S. Zhang, X. Dong, and H. Yang, "Designing states, actions, and rewards for using pomdp in session search," in *Advances in Information Retrieval* (A. Hanbury, G. Kazai, A. Rauber, and N. Fuhr, eds.), pp. 526–537, Springer International Publishing, 2015.

[8] N. Roy, G. Gordon, and S. Thrun, "Finding approximate pomdp solutions through belief compression," *Journal of Artificial Intelligence Research*, vol. 23, p. 1–40, Jan 2005.

[9] Y. Zhang, *Real-time SLAM for Humanoid Robot Navigation Using Augmented Reality*. PhD thesis, Applied Sciences: School of Mechatronic Systems Engineering, 2014.

[10] M. Ocana, L. M. Bergasa, M. A. Sotelo, and R. Flores, "Indoor robot navigation using a pomdp based on wifi and ultrasound observations," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2592–2597, Aug 2005.

[11] S. Iijima and I. Kobayashi, "A pomdp-based multimodal interaction system using a humanoid robot," in *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Posters*, pp. 519–525, 2016.

[12] R. Kaplow, A. Atrash, and J. Pineau, "Variable resolution decomposition for robotic navigation under a pomdp framework," in *2010 IEEE International Conference on Robotics and Automation*, pp. 369–376, 2010.

[13] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM national conference*, pp. 517–524, 1968.

What is this paper about?

Barták, Roman
Charles University
bartak@ktiml.mff.cuni.cz

Estimated Utility of:
"Trained Probabilistic Models for the NAO Robot in a Labyrinth"
Utility to Community: \$1
Utility to Society: \$1

The paper is about learning probabilistic models for NAO robots, at least, this is how I understand it. Unfortunately, the paper is very unclear about what the authors want to achieve, the problem solved is never specified precisely, it is not clear from the text, what the authors are doing and there is no justification. I strongly suggest the authors to think about what the reader is supposed to take away from the paper and I believe the paper requires complete rewriting.

As I understand, the authors are suggesting a Dynamic decision network to describe behaviour of Nao robots. This network is very high level with nodes likes location and angle (I would even say a textbook example), it must be clearly said how this is connected to Nao robot. It could be used for any moving robot, so what is specific about Nao there? The authors must clearly describe what sensor readings they use and what actuators they use. The authors wrote that Nao cannot autonomously and robustly walk. Is this really true? Isn't Nao used in RoboCup? How is the labyrinth used in the research? There was nothing said about the labyrinth in the paper. About the DBN model itself, it is never said, if it it discrete or continuos. There are many repetitive pictures of parts of the network, isn't one picture enough? Are you learning the CPTs or are you doing filtering with DBN or what? How is reward calculated? How do you use POMDP there? The section on related works is very unclear, how is multimodal interaction related to the paper? Section 5 on Experiments does not describe any experiment, no results are presented there. How did you evaluate your results? What is Section 5.3 supposed to say?

Sorry, I do not really understand what the contribution of this paper is.

Rebuttal on behalf of Trained Probabilistic Models for the NAO Robot in a Labyrinth

Samantha Eaton, Kleanthis Zisis Tegos,
Sagar Vilas Jagtap, Akram Alghanmi, Godwyll Aikins,
Neda Keivandarian, Ran Bi, Jesse Torres, Kholud Alghamdi, Marius Silaghi
Florida Tech
`class.project.2020S@fit.edu`

Answer To Review:
"What is this paper about?"
Utility Review: $200

Here we address in detail the items suggested by the author of the review entitled "What is this paper about?".

Since the reviewer has raise multiple independent questions in a single paragraph, we will split our rebuttal in separate sections, one for each sentence in the question.

The paper is about learning probabilistic models for NAO robots, at least, this is how I understand it.

The reviewer has correctly understood the main purpose of this research. Some in the FCRAR community might not yet be familiar with the importance of these models, and this review is very useful in providing us with clues concerning how to introduce the community with the appropriate background they need to understand the direction of some of the most advanced artificial intelligence research for handling uncertainty in robotics.

Unfortunately, the paper is very unclear about what the authors want to achieve, the problem solved is never specified precisely, it is not clear from the text, what the authors are doing and there is no justification.

While the original submission does not directly address what specific problems our models try to solve, it cites the previous FCRAR publications which describe these problems.

Shortly, our work aims to build a probabilistic model of the NAO robot's sensors and actions non-determinism, enabling the application of high-level intelligent algorithms for tasks in a vynil labyrinth.

The tasks context include school maze problem solving contests for educating students in advanced approaches to artificial intelligence concepts like uncertainty and non-determinism.

The foreseen contest application is the navigation of the maze while making educated movement decisions based on the information available. Decision components can include operations such as localization, mapping, and planning.

I strongly suggest the authors to think about what the reader is supposed to take away from the paper and I believe the paper requires complete rewriting.

We assumed that the target community, consisting of many of the attendees of FLAIRS, have seen the cited presentations of earlier parts of this work in the 2018 and 2019 editions, where the goals of this research was presented.

Thank you for point about the need for self-containment concerning motivations. To ease the introduction of other readers we will make this report more self-contained by inserting in the camera ready version the suggested section on motivation target applications, which we believe will alleviate the background difficulties described by the reviewer in this question.

As I understand, the authors are suggesting a Dynamic decision network to describe behaviour of Nao robots.

Indeed, we show a Dynamic Decision Network (DDN), proposing a Partially Observable Markov Decision Process (POMDP) model adapted to the NAO robot-maze architecture and whose conditional probability tables are trained for the vynil labyrinth environment based on extensive measurements in a provided database, also enriched by the data gathered with the experiments in this paper.

This network is very high level with nodes likes location and angle (I would even say a textbook example), it must be clearly said how this is connected to Nao robot. It could be used for any moving robot, so what is specific about Nao there?

While this network's elegance might almost be a textbook example, what makes this implementation specific to NAO is the structure and the parameters that each node represents. The sonar sensors are placed symmetrically right and left on the NAO's chest and therefore, the sonar data will only depend on the location and orientation of the robot's torso. The landmark node utilizes visual data from the cameras placed one on top of the other on NAO's head. Because the parameters that the network makes use of are highly dependent on their sensor's physical placement on the robot's body, the model and its implementation that we present to describe the robot behavior is unique to the NAO robot.

If this network was to be generalized for use, for instance, on another robot with cameras that the landmark node draws data from, fixed to the robot's torso; it would completely invalidate the network as the landmark node would not depend anymore on the head angles. It is however possible to use this network on another robot that shares the same physical features and sensors as NAO or with appropriate adaptation for robots that share similar features.

We are honored by the reviewer's estimation that that our solution is almost a textbook example, considering that this witnesses to the elegance of our approach, even as it is able to model all relevant aspects of the Nao problem.

The authors must clearly describe what sensor readings they use and what actuators they use.

We can now fathom that for readers not experienced with Nao it seems incredible that that the only relevant sensors are the two Sonars and the Landmark detector, for each of which our original submission has multiple sections allocated. The probabilistic analysis of these sensors is actually what our probabilistic models do model.

Similarly the only actuator exploited is the `walk` command, whose analysis and probabilistic model is the subject of the second part of Section 4.1.

The sensors were discussed in Section 2.1. The structures of their values are described in Section 2.4. Their logical representation is the main subject of Section 3. The analysis of these two sensors and of the actuator were the main subject of Section 4.1. The study of these sensors and actuators was one of the main experimental contributions of the paper, presented in the Experiments section, Section 5.2.

We thank for the reviewer's interest in sensors and actuators, which is a testimony to the importance of our article. And, while it does happen one "*can't see the forest for the trees*," we will improve the description by repeatedly pointing to the completeness of our study, to clarify the reader unfamiliar with Nao.

The authors wrote that Nao cannot autonomously and robustly walk. Is this really true? Isn't Nao used in RoboCup?

The NAO robot has an intrinsic error in its walking movement that has been noticed through experimentation, which on significant distances accumulates and causes unexpected behaviors.

This has also been noticed during our experimentation and this claim can also be backed by the physical sensor data that this paper uses which was gathered by a team whose 2019 article [2] we have cited. For instance, in the transition data obtained from their team, when the robot was asked to go to state 0 from state 4, in a square of 9 states defined in their paper, the robot ended up in state 0 only 63% of the time. Because of these uncertainties precisely, there is a lack of software support for high level intelligence with regards to locomotion. A probabilistic model-based approach can account for these uncertainties and enable us to create a more robust software infrastructure to integrate high level intelligence techniques for more accurate locomotion while performing localization.

One of the contributions of the current article was to quantify through analysis of experiments, in the form of probabilistic models, the otherwise known fact that Nao's walking actuators are non-deterministic. It also checked how well the Nao database can be used for this quantification.

We confirmed through experimentation, reported in prior work, that the NAO robot can only walk on special surfaces. We found out that the NAO could not properly walk on soft surfaces (e.g., carpet); the robot oscillates and then tips over. Even on special surfaces, turns and displacements are far from exact. For example, if the robot is given a command to turn \$x\$ degrees, it is more likely than not that it undershoots, overshoots, or does not move at all.

Proper software must be made for the NAO robot to be able to compensate for these deficiencies.

On turns of at least 10 degrees the turn becomes more robust, but inexact in value. On a command to turn 10 degrees, Nao generally turns $11\frac{1}{4}$ degrees.

As for RoboCup, the robots are autonomous but,

- Their techniques are not available to general Nao owners like us, but are restricted to a proprietary operating system only available to participants of the RoboCup. The RoboCup robot's developers get access to a special Naoqi image root, which grants them extra functions.

- Based on our understanding of the published procedures, their current approaches are low-level reflexes and do not have high-level intelligence. As we describe in our original submission, currently they are only usable for the specific tasks/roles like kicking and following the balls in the competition, rather than utility based reasoning and acting.

As cited in our related work, there are successful applications of Simultaneous Localization and Mapping (SLAM) to perform task like wall following in mazes and localization in spaces filled with landmarks. However, there is a lack of general software packages applicable to new scenarios so, our work will provide a solution towards designing general mapping and localization support software architecture with Nao.

How is the labyrinth used in the research? There was nothing said about the labyrinth in the paper.

The makeup of the labyrinth is talked about in the introduction section of the paper, second paragraph, and the cited article [1] is dedicated to its design.

All the landmark measurements are based on the positions of NaoMarks in the labyrinth so, inherently, the conditional probability tables (CPTs) of the computed model are only relevant in the context of the given labyrinth.

The readings of the sonar sensors also depend on the labyrinth wall material. Different materials reflect the sonar waves differently. Therefore its CPTs are also relevant only within this particular setting.

The labyrinth also provides the testing environment for the sample models test application described in experiments. Measurements of the sensors were extracted at various locations as described in the previous team's article, [2], measurements which were used to train the models described in this paper. The labyrinth was used to measure the accuracy of the obtained models, in terms of number of times the network guessed the current location correctly, given its sensor measurements.

About the DBN model itself, it is never said, if it it discrete or continuous.

The DBN that we created is discrete. It is explained in Section 4.1, where the discrete representation is given in detail, listed in tables, with domains quantified symbolically.

While location is a continuous quantity, the state space was divided into discrete cells. The same approach was used for the sensor measurements, which are also continuous, but were categorized into discrete groups.

There are many repetitive pictures of parts of the network, isn't one picture enough?

Our paper presented two different ways of evaluating the problem; one where the location is split into position and heading, and one where the location is not split. Pictures of these two DDNs could have been sufficient for a conference dedicated only to advanced AI, but the other figures were provided with the pedagogical goal of introducing the various DDN components to the FCRAR community, which is also composed of a mechanical engineering population segment assumed less familiar with DDNs. The additional images showing the incremental development are also an opportunity to introduce the CPTs involved in them without interference from the other components.

Are you learning the CPTs or are you doing filtering with DBN or what?

The CPTs are trained from the available databases and newly contributed experiments, as discussed in the Concepts and Data Structures section number 3 of our paper, and later in the experiments section, Section 5.

The Experiments section 5.3 contains a sample use of the created models with a filtering algorithm application, as a test of the contributed framework.

How is reward calculated?

The rewards needs not be calculated by us, as it is supposed to be the way for end-users of our library to specify tasks. This is similar to how queries are not included in a first order logic knowledge database, but are posed by the end users. So, while we here provide an AI library of probabilistic models, the end-users of such models sets the rewards for achieving specific tasks.

For example, in some designs the reward for the POMDP could be proportional to the distance of a cell to any of the surrounding walls, as well as the expected distance to the end position. What this means is that the robot will try to choose the move that will bring the robot closer to the end goal, taking the walls into consideration.

How do you use POMDP there?

As for the utilization of Partial Observable Markov Decision Processes (POMDP), the Dynamic Decision Network (DDN) is the graphical representation of POMDPs. The images that the reviewer questioned earlier are about the incremental construction of the POMDP from components. We did expect that the reviewers and readers in the conference may not be familiar with POMDPs and this is why we spelled graphically its construction using DDNs.

The solution of a DDN is typically using POMDP algorithms.

The section on related works is very unclear, how is multimodal interaction related to the paper?

The third described related article is about the multimodal interaction system between a human and a Pepper robot, using the POMDP framework. Pepper is a humanoid robot like Nao. It shows that POMDPs were previously used for emotive behaviors, even if not for walking, with such robots. Further, in the future, theoretically we could investigate the interaction of the NAO with more complex sound and vision inputs.

Section 5 on Experiments does not describe any experiment, no results are presented there. How did you evaluate your results?

The experiment consisted in measuring and evaluating sensor inputs for more situations, and analyzing the data into CPTs as reported in the article for better estimation of the scale of non-determinism. The CPTs, and related software for access, are too large for reproduction in the article but they are made available in the database. Snippets were provided in various figures. The results were indeed distributed over many sections of the article, and the summary in the Experiments section will be made more comprehensive.

What is Section 5.3 supposed to say?

Section 5.3 is the reference to the code provided with our experiments. We focused on providing a platform on top of which other researchers could improve upon in the pursuit for high level walking tasks in the maze.

Sorry, I do not really understand what the contribution of this paper is.

As described in the rebuttals to the previous detailed questions of the reviewer, our constructed probabilistic models, that the reviewer acknowledged in his first statement, is an architecture of library components for high level artificial intelligence behaviors. These components are usable for end-users implementing specific tasks in a vynil labyrinths, as exemplified by the sample simple pedagogical application in the Section 5.3 in our article.

As also mentioned in our cited earlier work [1], the targeted tasks that can be supported by our high level artificial intelligence libraries include school maze problem solving contests for educating students in advanced approaches to artificial intelligence concepts like uncertainty and non-determinism.

Thus we contribute to understanding the behavior of the Nao in a restricted environment, in addition to generating a large amount of data which can be utilized by researchers for training models with a similar environment.

We thank the reviewer for considering that the elegance of our solution is textbook level, even as it describes all the complexities of the Nao navigation problem in vynil mazes. We think that the insertion in the final version of the motivational applications and clarifications that the reviewer suggested, as we describe in this rebuttal, will indeed make the article also readable to a more general public.

A Neurofeedback-Driven Humanoid to support Deep Work

Troy R Weekes and Thomas C Eskridge

L3Harris Institute for Assured Information
Learning, Abstraction, Similarity and Reasoning (LASR) Lab
Department of Human Centered Design
Florida Institute of Technology

ABSTRACT

High performance is desired in the workplace, even with swarms of robots on their way in the Fourth Industrial Revolution. Our research focuses on the population of knowledge workers, who are typically expected to sit in one space for extended periods while performing deep, intellectual and creative work. For those who work predominantly using computers, there is growing scope to augment task performance using artificial virtual agents. This trend is evident in the adoption of voice-based, and gesture-based applications that allow users to issue vocal or gestural commands while their hands are occupied on primary tasks. Even though multimodal interaction may yield more productivity than solely mouse and keyboard interactions, it may still impose a significant cognitive load on the user. We propose the modeling of a smart motivational humanoid assistant that is personalized to interact with human users without explicit commands, and instead via wireless sensors that can perceive the operator's brain activity. The humanoid engages with the human using effective nudges through neurofeedback.

Keywords: Robotics, Neurofeedback, Deep Work, Nudges

Figure 1. Proposed context of use shows Nao humanoid interacting with a knowledge worker completing work tasks using a computer in the office workspace. Image courtesy Troy Weekes.

1. Introduction

Knowledge workers perform "non-routine cognitive jobs" whose ideas and concepts are responsible for the growth and development of businesses around the world. In the US, knowledge workers constitute almost 50% of the workforce [1]. They work with tools and automation that relieve them of many tasks, enabling them to deeply immerse themselves into the difficult knowledge tasks that cannot be automated [2]. The research question arises, how should human workers interact with complex automated technologies, which is itself a knowledge-work task? And, can this be done in a way that enhances deep cognitive work?

While the work environments of knowledge workers vary widely across different industries, this paper focuses on the office workspace and to the knowledge worker operating a computer system on a desk to complete a range of work activities. The primary artifacts in this context are the computer, and the top surface of the desk as shown in Figure 1. The knowledge worker may be a novice or an expert who may be succeeding on tasks, making errors, overloaded or distracted. Tasks may vary from writing documents, responding to email, computing calculations or debugging software code. Secondary artifacts in the office space such as lamps, toys, books, posters and windows may be used for changing mood, taking a break or sparking creativity.

The organization includes all supervisors and team-mates who interact with the knowledge worker to perform work tasks using the artifacts. Situations may be normal, abnormal or emergency scenarios that determine the relevant interactions during operation time. Situation complexity is often caused by interruptions, context-switching, and high workload conditions.

One predominant problem in environments such as these is the propensity for the knowledge worker becoming distracted to the detriment of work completion. Due to the reduced attention or impaired alertness, the inevitable net result is increased human error, and reduced ability to work safely, and productively [3] and decreased job satisfaction [4]. Human-centered design of the humanoid robot as a collaborative team-mate may enhance the task performance and psychological well-being of the knowledge worker. The robot should not cause stress and discomfort but incorporate social signals, cues and norms into the planning and control architecture to ensure psychological safety of the knowledge worker [5].

In this paper, we present the first steps of research into developing a computational model for a humanoid robot to support the cognitive needs of knowledge workers by increasing the duration and depth of their cognitive work. The second section of this paper outlines the state-of-the-art research on the psychological state of flow, neurofeedback-driven nudges, and human-robot interaction. In the third section, we present a design rationale for the smart personalized humanoid assistant. A demonstration of the agent-based framework was conducted to analyze the effects of the neurofeedback processing, and action planning modules. In the final section of the paper, we discuss the implications of neurofeedback-driven nudges, and conclude with limitations of our research, and suggestions for future work.

2. Related Work

2.1 Deep Work and Flow State

Knowledge workers are exposed to both internal and external distractions and interruptions that lead to suboptimal productivity, increased stress and dissatisfaction with their work. The average knowledge worker spends 30 percent of their daily time on email, and another 30 percent on team communication and Internet activities [2]. A significant portion of lost time is spent in context-switching, where additional time is lost before an interrupting task is started, and after it ends [6].

Distractions between episodes of knowledge work are a significant source of lost productivity and dissatisfaction due to the difficulty in accomplishing work goals. If these periods of distraction can be decreased or eliminated, the resulting ability to focus and complete tasks may positively impact on productivity.

Flow is defined as "the state in which people are so involved in an activity that nothing else seems to matter; the experience itself is so enjoyable that people will do it even at great cost, for the sheer sake of doing it" [7]. In the workplace, flow occurs when individuals, teams or organizations operate with optimal focus, and perform without apparent effort, which yields a heightened sense of motivation, intrinsic satisfaction, and peak outcomes [8, 9].

Flow experiences are not always easy to attain and sustain, but developing the ability to control attention may be an effective way to find and maintain flow [10]. Concentration is a significant component of achieving flow [11]. To establish the flow state, the human must focus attention on the activity, and goals at hand. Our hypothesis is that if the operator is aware of their state of concentration, they will be better able to drive towards deeper levels.

2.2 Neurofeedback and Nudges

Attention focus is a trainable skill that may result in reaching the flow state [12]. One way to train sustained attention is to provide a sensitive feedback signal so human users can learn to sense upcoming lapses earlier and prevent them from occurring in behavior [13]. Neurofeedback training has produced outcome gains in sustained attention [14], and has been used for the purpose of cognitive enhancement and as a therapeutic tool [15].

Neurofeedback is a category of biofeedback that is based on brain electrical signals recorded by bio-sensors placed on the head of humans. Neurofeedback involves the collection of the brain signals, classification and presentation of a stimulus to make individuals aware of their brain states and how to self-regulate certain brain-based processes [16]. In general, effective use of biofeedback requires specialized equipment to convert physiological signals into meaningful cues or a trained biofeedback practitioner [17].

To be effective working with the most difficult knowledge tasks in a domain, human workers often rely on external aids such as rituals, reminders and mental hacks, which this work collectively calls "nudges". This set of work-related nudges are used mostly on an ad-hoc basis to propel workers to deep levels of concentration and achievement. Nudges are used in behavior science to positively influence people while preserving their freedom of choice, and without engaging their deliberative capacities [18, 19].

The purpose of the reinforcement learning loop shown in Figure 2 is to help the knowledge worker achieve, and maintain a flow state while performing tasks. The aim is to have the Nao humanoid monitor and mirror the human's state so that it can effectively motivate them with nudges when they drift off task, and become invisible when their task performance is in the flow state.

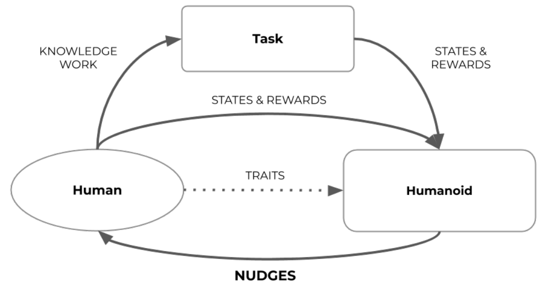


Figure 2. The reinforcement learning control model depicts the human completing work tasks in an operational environment that allows interactions with the humanoid agent. Percepts of human state and traits represent the agent's observations over time. The reward signal is obtained based on the humanoid's likeness to the human's subsequent flow state, and task completion. The humanoid performs nudge actions that consistently target stimulation towards deep flow.

The proposed nudges incorporate sound, physical motion, and other visual cues, and aims to combat boredom, fatigue, anxiety, distractions and habitual behaviors that may lead to under-performance during knowledge work. The nudges are delivered using traits of transparency and expressiveness versus autonomy and accuracy [20], good automation etiquette versus socially-agnostic behavior [21], and empathetic verbal apology over sterile error codes [22, 23].

Andujar et al [24] proposed that adapting humorous smart technologies in the workplace can be beneficial for employee retention and the improvement of employees' positive emotional state. The designers of humanoid interactions are recommended to consider how to detect negative responses resulting from the uncanny valley effect (where people react with unease seeing humanoid behavior that closely, but not perfectly, matches human behavior), and design robot behaviors to mitigate this by using humor or apology to overcome the robot's disagreeable appearance [25, 26, 27].

2.3 Cognitive and Collaborative Robots

Cognitive robotics involves the use of bio-inspired methods for the design of sensorimotor, cognitive, and social capabilities in autonomous robots [28]. Reggia et al [29] designed a cognitive humanoid robot framework with components that center on top-down control of a working memory that retains explanatory interpretations that the robot constructs during learning. The core function of our humanoid robot is to learn about the mental and emotional states of the knowledge worker and respond in a way to support their cognitive needs.

Collaborative robots are designed to physically interact with humans in a shared workspace instead of replacing them [30]. The humanoid robot proposed in this research provides personalized interacts with different knowledge workers while conducting work tasks in their workspaces. This interaction model is non-competitive cooperation by mutual understanding where authority is traded between human, and collaborative robot [31].

The sense of presence, and enjoyment that people feel with a robot can be manipulated by changing its social abilities [32]. In particular, the aesthetics, functionality, embodiment, situatedness, and morphology of humanoid robots can affect its interactions [33]. For example, Jo et al [34], found that interactions with physical and virtual humanoid robots produced a statistically significant effect in increasing human creativity. Automation inaccuracy can degrade human trust, comfort, and acceptance [35]. For these reasons, the humanoid's expressions and feedback are designed as a clear imitation of the knowledge worker's flow state.

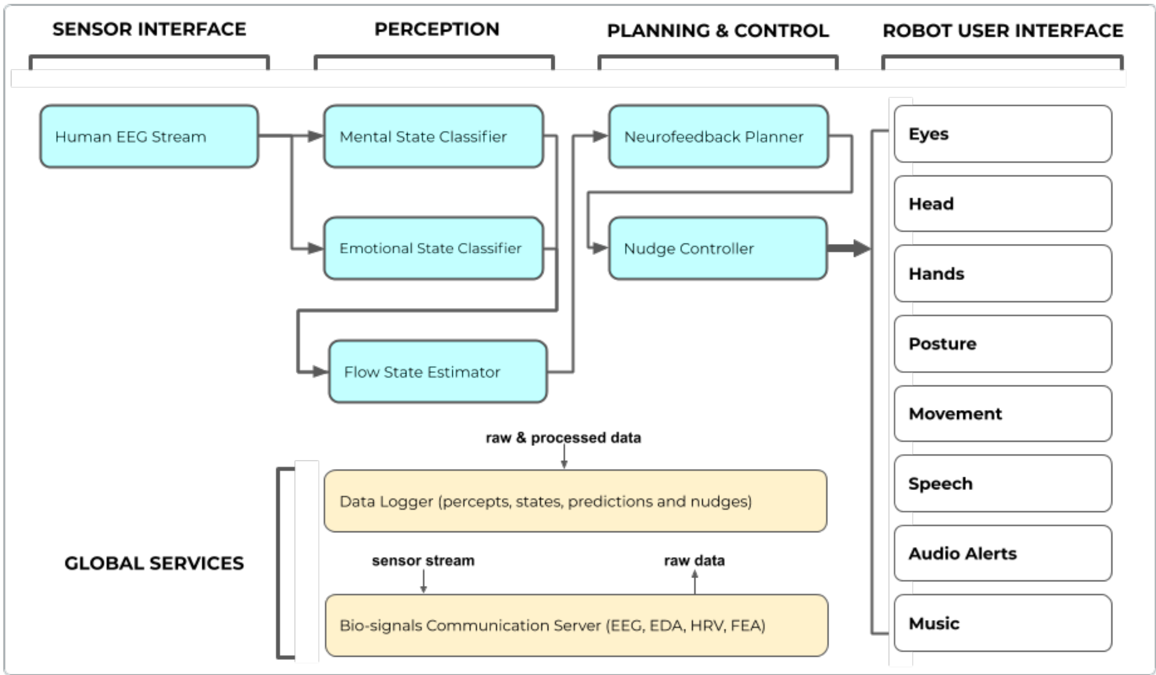


Figure 3. Pipeline architecture of the neurofeedback-driven humanoid where the modules process data on server and client threads in parallel.

3. Deep Work Humanoid Implementation

The humanoid obtains its bio-sensing capability from modules that process raw brain sensor data, classify human state, and plan appropriate nudges. The humanoid listens to the nudge controller, and executes nudges through expressive actions with its effectors.

3.1 Human EEG Stream

The human electroencephalogram (EEG) stream is collected from four EEG electrode bio-sensors on the Muse headband [36]. The sensor locations using the standard 10-20 coordinates are analogous to the prefrontal regions (AF7 and AF8) and the temporal regions (TP9 and TP10). The raw EEG data are sampled at 220 Hz on the four channels. The raw EEG data are transported to the server database via a mobile device interface that is capable of being synchronized with motion and other bio-signals.

3.2 Mental and Emotional State Classifiers

Noise and artifacts such as blinks and jaw clenches are filtered from the raw EEG data using a bandpass filter between 1 Hz to 75 Hz. The filtered EEG data are segmented into 4-second epochs without overlap. A short-term Fourier transform on each segment produces power spectral density (PSD) features in 5 frequency bands (delta, theta, alpha, beta and gamma). PSD features are classified using a recurrent neural network (RNN) with long short-term memory (LSTM) units. The mental states are engaged or disengaged. The emotional states are bored (B), happy (H) or anxious (A).

3.3 Flow State Estimator

In our research, the flow concept is operationalized with two dimensions: human skill and task challenge. Flow is modeled as the region where there is a balance between the human skill and task challenge. Flow is estimated as a discrete state ($\text{Distracted}_\text{B}$, $\text{Shallow Flow}_\text{B}$, $\text{Distracted}_\text{A}$, $\text{Shallow Flow}_\text{A}$, $\text{Shallow Flow}_\text{H}$ and $\text{Deep Flow}_\text{H}$) measured 4 seconds after presentation of the nudge.

3.4 Neurofeedback Planner

A common approximation to reinforcement learning (RL) problems is to ignore noise, which assumes full observability by learning and planning in observation space rather than a latent state space [37]. However, such approximations break down when using the Nao robot due to non-determinism in the actuators [38], and the EEG signals are stochastic due to sensors noise and artifacts. Therefore, the RL control model in Figure 2 is a discrete-time partially observable Markov decision process (POMDP). A Q-Learning algorithm is used to obtain the policy that maximizes the expected sum of rewards. The policy is used to plan and map the optimal nudge to the estimated flow state of the knowledge worker.

3.5 Nudge Controller

In order to help the knowledge worker maximize on deep work, the humanoid nudges with gestural feedback, reminders of task goals, and, in some cases, modifies the challenge level of the task relative to the skills of the knowledge worker. Nudges are conveyed in the form of expressive actions that the robot executes based on probabilistic computations of percepts, predictions and rewards. These humanoid expressions that are shown in Figure 4 represent stimuli to deliberately influence the knowledge worker by mimicking their mental and emotional state while motivating transition to deep flow.

3.6 Global Services

The architecture has a set of global services for logging percepts of raw data, PSD features, classified states, predictions and nudges. Streams of bio-sensor packets are transferred and aggregated via Bluetooth and User Datagram Protocol (UDP). The bio-sensor packets are synchronized and logged with discrete timestamps. Nudges are communicated from the server over Wi-Fi to the robot.

3.7 Robot User Interface

Only dependable robot architectures can be accepted for supporting “human-in-the-loop” conditions and human–robot teams [39]. Humanoids are a special case because they intrinsically present multiple control points for grasping, moving the head for perception, assuming postures, walking, and so on [40]. We hypothesize that a tangible interactive robot with the form factor of a warm, competent, and sensitive humanoid will be acceptable, and effective in knowledge work scenarios. The Nao humanoid supports the required human-like motor capability and has a high level of expressiveness that is capable of delivering a variety of nudges in Figure 4, which makes it suitable for our human-robot teamwork function. According best practices for social robots [26, 41, 42], we designed the humanoid to exhibit the following social characteristics:

- perceive, and express emotions;
- communicate with body language, and sound;
- exhibit a distinctive personality, and character;
- perceive natural cues (gaze, facial expression, head posture).

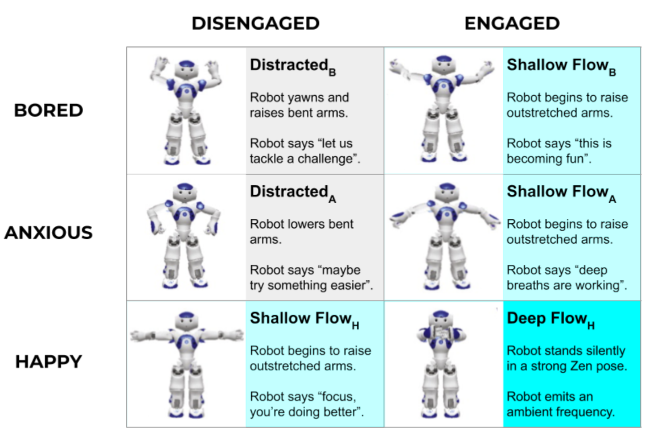


Figure 4. The six nudge categories of the Nao robot represent a sparse encoding that correlates expressions to the flow state, which is a product of mental and emotional state of the knowledge worker.

The expressions of the robot in Figure 4 are timely interactions with the knowledge worker without overt commands. These expressions strategically auto-encode the flow state of the knowledge worker into the metaphor of a Zen practitioner. The fluid movements of the humanoid are coupled with silence, music or motivational phrases spoken by the robot. Within the six categories of expressions, there is granularity of the phrases and sounds to improve salience and mitigate monotony. Silent motions avoid distracting the knowledge worker unnecessarily, and simple phrases minimize the cognitive load required to understand the robot.

According to Breazeal [41], the importance of feedback, and the readability of expression in this process cannot be underestimated in human-robot interaction. As the human applies the social model to understand the robot, they are constantly observing the robot’s behavior, and manner of expression to infer its internal states. This allows the person to predict, and understand the robot’s behavior only if the robot’s expression is readable (the intended signal is appropriately interpreted by the human). The robot’s expression reliably maps to the internal state being expressed, and this internal state adheres to the human’s mental model of the robot.

4. Demonstration

In this section we review a demonstration of the architecture, showing how the pipeline architecture ingests classified mental and emotional states from EEG data, and then decides which nudge to produce on the humanoid robot. The synthetic dataset used in this demonstration simulated 2,000 trials i.e 100 work-hours. The trials were randomly sampled from 18 types of knowledge work scenarios. Each scenario type exhibited an overarching emotional state e.g. boring, anxious or happy. In order to simulate different frequencies of distractions, the scenarios were combined with the following types of perturbations in mental state:

- Engaged for 3 minutes
- Disengaged for 3 minutes
- Engaged for 90 seconds then disengaged for 90 seconds
- Disengaged for 90 seconds then engaged for 90 seconds
- Engaged then disengaged alternating every 1 minute
- Engaged then disengaged alternating every 12 seconds

The 3-minute scenario graphically depicted in Figure 5 shows a situation where a knowledge worker is in an overarching happy emotional state but experiencing intermittent 12-second distractions. Given this processed data from the humanoid’s sensor and perception layer, the resultant flow state is handed to the neurofeedback planning and nudge control layers then an expression is communicated to the robot’s user interface.

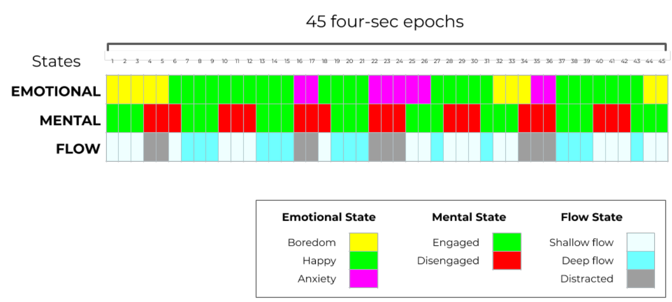


Figure 5. Visualization of a knowledge work scenario showing the emotional, mental and flow states over 45 epochs of sensor data after pre-processing and classification in the humanoid’s software pipeline.

We adopted a Q-Learning approach as the baseline reinforcement learning (RL) control method for the neurofeedback planner. The RL method was implemented in Python to approximate the action-value function of the optimal policy. The Q-Learning algorithm used temporal differences to update the scores its Q-Table.

The data was split 50:50 into training and test sets, and reused across all experimental cases. The ε–greedy algorithm was used to make use of the exploration-exploitation tradeoff, and a linear function was used to decrement ε over the scenarios. Three reward functions were used to generate the different policies. The state-based function calculated reward based on the value of the resulting state if it was predicted. The distance-based reward function used the distance between the resulting state and the predicted state. The combo function represented a weighted combination of both measures. The nudges from the neurofeedback planner were tested on the Nao humanoid via the nudge controller. The following tables summarize the nudges resulting from the computational model for discussion.

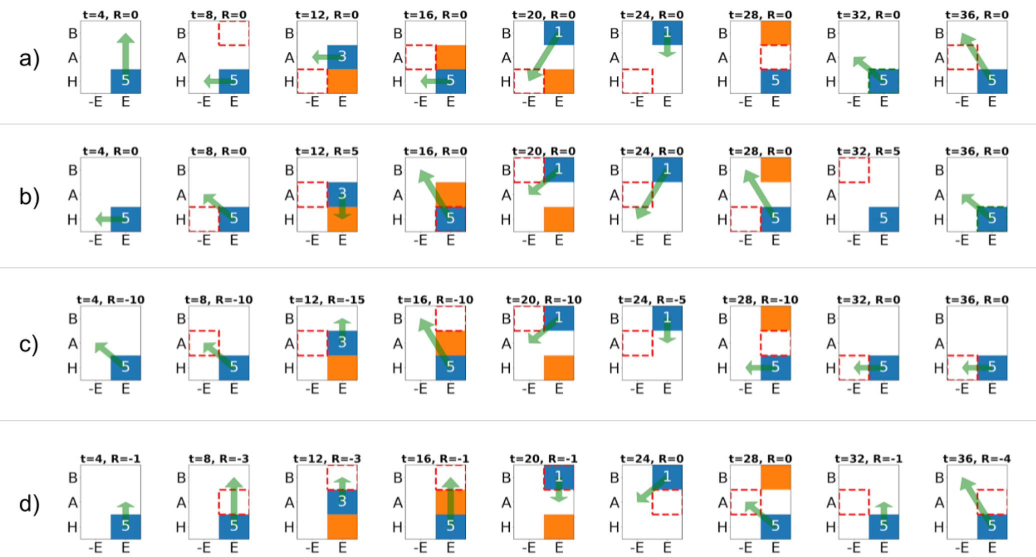


Figure 6. Comparison of policies for knowledge work scenarios across 9 four-sec epochs. The plots represent (a) before the humanoid was trained, and after training with (b) state-based reward, (c) distance-based reward, and (d) a combined state- and distance-based reward. The green arrows depict the attempted nudge. The blue cell represents the current observation, and the orange cell represents the previous observation. The dashed square highlights the previous nudge, where red indicates an incorrect prediction and green indicates a correct prediction.

4.1 Discussion

Figure 6 shows a comparison of three policies that were learned by the agent based on three different reward functions. Nudges in the baseline (a) appeared to be random and independent of observations. The state-based reward function in section (b) shows a different set of nudges, which yielded low rewards on the predictions. The distance-based reward function in section (c) depicted a more constrained set of nudges, which were closely related to the current observation. The combo-based reward function demonstrated that it is possible to combine benefits from multiple reward signals.

The nudges used in this demonstration were delivered every 4 seconds during the given knowledge scenario. When executed on the Nao, even though some nudges were silent, the number of nudges was relatively highly across the scenarios on average. This is likely to make the humanoid a distraction to maintaining flow and deep work. When the humanoid is deployed, an alternative approach to implement the neurofeedback planner would be to only nudge after (1) a stabilized period of the flow state, (2) after dwelling in distraction for 10 seconds, and (3) after task completion. The nudge after completing the scenario may be planned by classifying the knowledge worker's flow performance over the scenario and recommending the next task from a task list based on its estimated duration and challenge relative to the knowledge worker's skill.

5. Conclusion

This main goal of this research was to develop a computational model that effectively applies human-robot interaction in the deep knowledge work domain to augment human performance. The architecture presented in this paper offers a lightweight framework for investigating biofeedback-sensitive aids to deep cognitive work. Our hope is that this framework lowers the barrier for future research into the efficacy of different nudge regimes.

Breazeal, Aly, and others are adamant that the long-term emotional effects of humanoid interactions on humans need to be investigated [43, 44]. Issues arising of ethical concern include over-reliance on automation [45], and potential manipulation of people through social robots [46]. Some AI algorithms may suffer from inaccuracies in the training data or may possess vulnerabilities to errors, and biased rules, which may lead to inaccurate profiling [47].

Notwithstanding the challenges and ethical issues, there is a possible design of the future where robots do not replace humans but enhance them to make a positive difference in the world. The impact of effective neurofeedback-driven humanoids that support deep work is potentially significant in terms of economic productivity, and social well-being of knowledge workers.

5.1 Future Work

Real human- and real humanoid-in-the-loop experiments are an exciting part of the next steps. Beyond the theoretical experiments reviewed in this paper, humanoids with simulation-trained neurofeedback planners will be compared with those trained with real world experience. Then we can determine if the humanoid's effectiveness may be improved by tailoring personalized nudges that fit the traits and preferences of specific knowledge workers.

Future studies will examine "attention" in greater detail, e.g. nudges that help knowledge workers to train their focus on task relevant stimuli, and boost their immunity to distraction. It is noted that a human can be focused and attentive on a distracting stimulus. This differentiation will require laboratory research with motion tracking to provide an objective measure of how attention is directed, and controlled by the knowledge worker during tasks.

Advanced work in feature engineering can improve the humanoid's accuracy and responsiveness. For example, the environment's reward signal that is currently controlled by task list completion and humanoid flow similarity may be augmented by human-centered metrics such as joy and positive affect, which may increase the reliability of the neurofeedback planner.

On the hardware side, the sensor interface may be extended to include different bio-signals such as facial expression, heart-beats, skin conductance, etc. The purpose of multimodal bio-signal fusion would be to increase classification accuracy, and improve the explainability of our deep learning models.

References

[1] J. Zumbrun, "The rise of knowledge workers is accelerating despite the threat of automation," *The Wall Street Journal*, May 4, 2016.

[2] C. Newport, *Deep Work: Rules For Focused Success in a Distracted World*. Hachette Uk, 2016.

[3] W. G. Sirois, "The myths & realities of fatigue: Reducing the costs, risks, and liabilities of fatigue in 24-hour operations," *Circadian White Paper*, 2009.

[4] A. Gazzaley and L. Rosen, *The Distracted Mind: Ancient Brains in a High-tech World*. Cambridge, Ma: Mit Press, 2016.

[5] J. Hu, B. Erdogan, K. Jiang, T. N. Bauer, and S. Liu, "Leader humility and team creativity: The role of team information sharing, psychological safety, and power distance.," *Journal of Applied Psychology*, vol. 103, no. 3, p. 313, 2018.

[6] S. Monsell, "Task switching," *Trends in Cognitive Sciences*, vol. 7, no. 3, pp. 134–140, 2003.

[7] M. Csikszentmihalyi, "Flow: The psychology of optimal performance," 1990.

[8] M. Csikszentmihalyi, "Play and intrinsic rewards.," *Journal of Humanistic Psychology*, 1975.

[9] J. Nakamura and M. Csikszentmihalyi, "The concept of flow," in *Flow and The Foundations of Positive Psychology*, pp. 239–263, Springer, 2014.

[10] M. Csikszentmihalyi, "Flow, the psychology of optimal experience, steps towards enhancing the quality of life.," 1991.

[11] M. Koufaris, "Applying the technology acceptance model and flow theory to online consumer behavior," *Information Systems Research*, vol. 13, no. 2, pp. 205–223, 2002.

[12] Y.-y. Tang and M. I. Posner, "Attention training and attention state training," *Trends in Cognitive Sciences*, vol. 13, no. 5, pp. 222–227, 2009.

[13] T. D. Megan, J. D. Cohen, R. F. Lee, K. A. Norman, and N. B. Turk-browne, "Closed-loop training of attention with real-time brain imaging," *Nature Neuroscience*, vol. 18, no. 3, p. 470, 2015.

[14] J. H. Gruzelier, "Eeg-neurofeedback for optimising performance. i: A review of cognitive and affective outcome in healthy participants," *Neuroscience & Biobehavioral Reviews*, vol. 44, pp. 124–141, 2014.

[15] S. Enriquez-geppert, R. J. Huster, and C. S. Herrmann, "Eeg-neurofeedback as a tool to modulate cognition and behavior: A review tutorial," *Frontiers in Human Neuroscience*, vol. 11, p. 51, 2017.

[16] J. S. Kreutzer, J. Deluca, and B. Caplan, *Encyclopedia of Clinical Neuropsychology*. Springerlink, 2018.

[17] D. L. Frank, L. Khorshid, J. F. Kiffer, C. S. Moravec, and M. G. Mckee, "Biofeedback in medicine: Who, when, why and how?," *Mental Health in Family Medicine*, vol. 7, no. 2, p. 85, 2010.

[18] R. H. Thaler and C. R. Sunstein, *Nudge: Improving Decisions About Health, Wealth, and Happiness*. Penguin, 2009.

[19] Y. Saghai, "Salvaging the concept of nudge," *Journal of Medical Ethics*, vol. 39, no. 8, pp. 487–493, 2013.

[20] M. Johnson, J. M. Bradshaw, P. Feltovich, C. Jonker, B. Van Riemsdijk, and M. Sierhuis, "Autonomy and interdependence in human-agent-robot teams," *IEEE Intelligent Systems*, vol. 27, no. 2, pp. 43–51, 2012.

[21] C. A. Miller, "Trust in adaptive automation: The role of etiquette in tuning trust via analogic and affective methods," in *Proceedings of The 1st International Conference on Augmented Cognition*, pp. 22–27, Citeseer, 2005.

[22] C. Ray, F. Mondada, and R. Siegwart, "What do people expect from robots?," in *Proceedings of The IEEE/Rsj 2008 International Conference on Intelligent Robots and Systems*, pp. 3816–3821, IEEE Press, 2008.

[23] Y. Iwamura, M. Shiomi, T. Kanda, H. Ishiguro, and N. Hagita, "Do elderly people prefer a conversational humanoid as a shopping assistant partner in supermarkets?," in *Proceedings of The 6th International Conference on Human-robot Interaction*, pp. 449–456, ACM, 2011.

[24] M. Andujar, A. Nijholt, and J. E. Gilbert, "Designing a humorous workplace: Improving and retaining employee's happiness," in *Advances in Affective and Pleasurable Design*, pp. 683–693, Springer, 2017.

[25] B. R. Duffy, "Anthropomorphism and the social robot," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 177–190, 2003.

[26] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 143–166, 2003.

[27] L. Huang, D. Mcdonald, and D. Gillan, "Exploration of human reactions to a humanoid robot in public stem education," in *Proceedings of The Human Factors and Ergonomics Society Annual Meeting*, vol. 61, pp. 1262–1266, Sage Publications Sage Ca: Los Angeles, Ca, 2017.

[28] N. M. Seel, *Encyclopedia of The Sciences of Learning*. Springer Science & Business Media, 2011.

[29] J. A. Reggia, G. E. Katz, and G. P. Davis, "Humanoid cognitive robots that learn by imitating: Implications for consciousness studies," *Frontiers in Robotics and Ai*, vol. 5, p. 1, 2018.

[30] J. E. Colgate, M. A. Peshkin, and W. Wannasuphoprasit, "Cobots: Robots for collaboration with human operators," in *Proceedings of The Asme Dynamic Systems and Control Division*, p. 433, American Society of Mechanical Engineers, 1996.

[31] G. Boy, *Orchestrating Human-centered Design*. Springer Science & Business Media, 2012.

[32] M. Heerink, B. KrÖSe, V. Evers, and B. Wielinga, "The influence of social presence on acceptance of a companion robot by older people," *Journal of Physical Agents*, 2008.

[33] B. Miller and D. Feil-seifer, "Embodiment, situatedness, and morphology for humanoid robots interacting with people," *Humanoid Robotics: A Reference*, pp. 1–23, 2016.

[34] D. Jo, J.-g. Lee, and K. C. Lee, "Empirical analysis of changes in human creativity in people who work with humanoid robots and their avatars," in *International Conference on Learning and Collaboration Technologies*, pp. 273–281, Springer, 2014.

[35] A. Hamacher, N. Bianchi-berthouze, A. G. Pipe, and K. Eder, "Believing in bert: Using expressive communication to enhance trust and counteract operational error in physical human-robot interaction," in *Robot and Human Interactive Communication (Ro-man), 2016 25th IEEE International Symposium On*, pp. 493–500, IEEE, 2016.

[36] P. Bashivan, I. Rish, and S. Heisig, "Mental state recognition via wearable eeg," 2016.

[37] M. L. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, no. 7553, pp. 445–451, 2015.

[38] A. Hasanain, T. Weekes, M. Person, K. Paul, Y. Chang, A. Rothman, Z. Rui, R. Rahil, M. Syed, C. Wodd, and Others, "Models and framework for supporting humanoid robot planning & exploration," 2018.

[39] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction," *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, 2008.

[40] A. De Santis, "Modelling and control for human-robot interaction: Physical and cognitive aspects," in *2008 IEEE International Conference on Robotics and Automation*, Citeseer, 2008.

[41] C. Breazeal, "Toward sociable robots," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 167–175, 2003.

[42] K. Dautenhahn, T. Fong, and I. Nourbakhsh, "A survey of socially interactive robots," 2014.

[43] A. Aly, S. Griffiths, and F. Stramandinoli, "Metrics and benchmarks in human-robot interaction: Recent advances in cognitive robotics," *Cognitive Systems Research*, vol. 43, pp. 313–323, 2017.

[44] C. Breazeal, "Emotion and sociable humanoid robots," *International Journal of Human-computer Studies*, vol. 59, no. 1-2, pp. 119–155, 2003.

[45] R. Parasuraman and V. Riley, "Humans and automation: Use, misuse, disuse, abuse," *Human Factors*, vol. 39, no. 2, pp. 230–253, 1997.

[46] R. Subramanian, "Emergent ai, social robots and the law: Security, privacy and policy issues," *Journal of International Technology and Information Management*, vol. 26, no. 3, pp. 81–105, 2017.

[47] T. Ring, "Privacy in peril: Is facial recognition going too far too fast?," *Biometric Technology Today*, vol. 2016, no. 7-8, pp. 7–11, 2016.

Review: A Neurofeedback-Driven Humanoid to support Deep Work

Filipovich, Roman
Apple
romanmail@apple.com

Estimated Utility of:
"A Neurofeedback-Driven Humanoid to support Deep Work"
Utility to Community: $50
Utility to Society: $50

This paper describes a protocol for optimizing performance of individuals working in front of computer screen. The paper proposes to use cues from on-the-desk humanoid to affect human performance. The input to the system is coming from EEG sensors mounted on individual's head. The optimization goal is tied to the concept of flow, which should correlate with stressless peak performance. Computationally, the problem is cast as DQN RL.

This paper is very timely given the recent events when many work in a similar setting from home and need to balance between productivity and health. Overall, the application is very useful and I am glad the authors submitted this paper.

Some of the things the authors may consider in the next iterations of this work: - It would be good to understand how the choice of robot action space impacts the performance of the overall system. Alternatively, having an action space that has clear motivation from psychology research would be nice. - Data processing and the choice of computational framework are sound, but it would be good to have a better justification for the choices made in the paper. - I agree with the authors' suggestion for the future work on real life experiments. It would be interesting to see it. - The system optimizes for short term success. It is important to consider mid- to long-term fatigue effects and analyze the approach from a long-term productivity perspective.

A Neurofeedback-Driven Humanoid to support Deep Work

Weekes, Troy
Florida Tech
tweekes1998@my.fit.edu

Answer To Review:
"Review: A Neurofeedback-Driven Humanoid to support Deep Work"
Utility Review: $50

1. Choice of action space and relation to psychology research

The action space of the humanoid is limited to the expressive actions that the Nao robot is capable of executing with its effectors. The nature of the subtle Zen-like movements and motivational phrases emphasize the salience of the action while not interrupting human's primary work tasks. These expressions represent stimuli that may influence knowledge workers to become aware of their mental and emotional states while motivating them to operate in the deep flow state. To this end, the action space includes expressions that are interpretable metaphors of the human's flow state. Apart from silence, which seeks to ensure that the human tolerates the robot, the nudges that were selected for the research are rooted in perceptual measurement (Chaudhuri, 2011) from psychophysiological research where the stimuli are discriminable among each other, and detectable from noise signals in the office context.

2. Justifications for the data processing and computational framework choice

In this paper, we focus on the neurofeedback planning and nudge controller aspects of the data processing and computational framework. The pipeline architecture of the framework facilitates the parallel execution of processes in the various layers. We then focussed on the components of the architecture which deal with sensation and perception. We assumed that the knowledge work scenarios in the synthetic dataset represent a valid sample of flow state sequences to train and test a Q-learning agent. The specific choice to demonstrate the Q-learning approach was to establish a baseline method from which future research and comparisons of reinforcement learning models will be pursued. The nudge controller was fitted to the Nao humanoid robot, which was selected because we have a unit that is accessible for use in real-world experiments.

The rationale behind the use of electroencephalography (EEG) as the basis for the neurofeedback source of the affective and cognitive state data is the temporal dynamics of the data streams, and the cost-effectiveness of the device. Other neuro-recording hardware such as Near Infrared Spectroscopy (NIRS), Functional Magnetic Resonance Imaging (fMRI) and Magnetoencephalography (MEG) are costly and require complex hardware setup. For the context of the office workspace, a light-weight and portable EEG headset provides accessible brainwave signals of reasonable quality for the state classification process. In relation to the fMRI, which has very high spatial resolution in its data, EEG has lower spatial resolution that is limited by the number of electrodes; however, EEG has much higher temporal resolution in its high-frequency data streams.

3. Optimizing for short and long-term success, and handling fatigue effects

To maximize rewards in a greedy approach, the humanoid may exploit its experience to select the nudge that appears best, and never truly explore the action space. This lack of exploration would prevent the humanoid from improving its knowledge about each of the nudges at the expense of short-term rewards because obtaining information about each nudge's payoff helps to refine their estimates, and improve long-term success.

Future nudge selection algorithms will include the history of the nudges, and contextual information about the human and nudges. Such approaches are likely to produce different policies in different preferences, e.g. "easy" vs "challenging" tasks, and different time horizons "3-minute" vs "3-hour" scenarios. When the agent learns the profile of the knowledge worker and their work, there will be a personalized context for which the optimal policy will be selected to determine the nudge at the given instant. Because the effects of the humanoid can be seen in the changes of operator state, this level of contextualization allows algorithms to account for situations where the knowledge worker may be experiencing humanoid fatigue and would wish for the robot to be totally silent and passive. With continued use, including these situations in the policy learning should help to make humanoid fatigue less likely.

When the knowledge worker interacts with the humanoid over time, it is also expected for the human to become aware of how their mental and emotional states are interpreted by the robot, and even predict the nudges. This would imply that the human has reached a stage of self-awareness where the neurofeedback may no longer be needed to help reach and maintain flow state.

Chaudhuri, A. (2011). Fundamentals of sensory perception. Oxford University Press.

Part III

Reviewed Article Drafts

3D Localization of Defects in Facility Inspection

Nicholas Califano
University of Central Florida

David Murakami
NASA Ames Research Center

ABSTRACT

Wind tunnels are crucial facilities that support the aerospace industry. However, these facilities are large, complex, and pose unique maintenance and inspection requirements. Manual inspections to identify defects such as cracks, missing fasteners, leaks, and foreign objects are important but labor and schedule intensive. Our goal is to utilize small Unmanned Aircraft Systems (sUAS) and computer vision-based analysis to automate the inspection of the interior and exterior of NASA's critical wind tunnel facilities. We detect missing fasteners as our defect class, and detect existing fasteners to provide potential future missing fastener sites for preventative maintenance. These detections are done on both 2D raw images and in 3D space to provide a visual reference and real world location to facilitate repairs. A dataset was created consisting of images taken along a grid-like pattern of an interior tunnel section in the AEDC National Full-Scale Aerodynamics Complex (NFAC) at NASA Ames Research Center. Our method uses object detection to create image level bounding boxes of the fasteners and missing fasteners, then uses photogrammetry to create a mapping from 2D image locations to 3D real world locations. The image level bounding boxes and the 2D to 3D mapping are then combined to determine the 3D location of the defects. We describe the data collection, photogrammetry, and computer vision techniques used for object detection as well as a quantitative analysis of the method.
Keywords: Small Unmanned Aircraft Systems (sUAS), Object Detection, 3D Localization.

1. Introduction

The NFAC facility is the largest wind tunnel in the world, and supports both military and commercial programs in the aerospace industry. Maintenance of the facility requires routine inspections to be performed by technicians, but it is nearly impossible to perform frequent, thorough, manual inspections of the NFAC facility as there is more than one million square feet of interior surface area. Additionally, the NFAC facility surfaces contain complex geometry and ceilings that vary in height from 50' to 250', making inspections of some areas particularly difficult.

Our goal is to use drones to capture images along grid-like patterns of the facility surfaces, and to automate the process of defect detection with computer vision-based analysis. The system must be able to provide the 3D location of the defect, as well as images containing the defect for visual reference. The proposed method in this paper outlines a pipeline to detect defects in 3D space from the set of images taken in the facility.

Figure 1: NFAC Facility at NASA Ames Research Center
Source: [1]

2. Related Works

Photogrammetry. Photogrammetry is a well-studied and accurate method to extract 3D spatial dimensions from a set of 2D images taken at various locations [2]. Through the use of classical computer vision methods, object recognition can be used to match local features between images. Given the camera poses, camera parameters, and optical parameters, the matched features can be used to determine the 3D location of the object [3]. Photogrammetry creates keypoints, each of which map a 3D point to the 2D pixel locations of the images in which that 3D point is visible. However, this method alone is not able to determine the class of the objects recognized and therefore cannot determine if a recognized object is a defect.

Object Detection. There are two main fields of computer vision that determine the pixel location of an object instance in an image: object detection and instance segmentation. Object detection creates a bounding box around each object instance while instance segmentation creates a pixelwise mask of each object instance. Since the application of wind tunnel facility inspections is relatively niche, a custom dataset was required. Manually labeling datasets is expensive in terms of monetary cost, time, and effort [4, 5]. Bounding box labels can be created using significantly less man hours than pixelwise masks. Additionally, bounding box level detections are more than sufficient for visual reference and to perform 3D localization. As such, object detection was selected as the preferred method.

All object detection methods are composed of three main components: a region proposer, a feature extractor, and a classifier. The

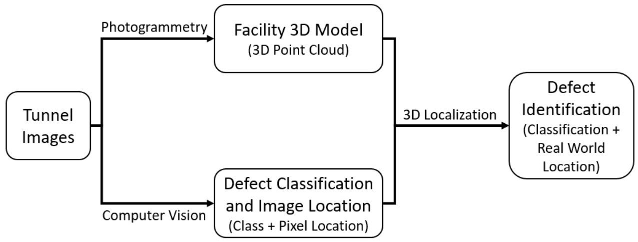


Figure 2: Method Pipeline

region proposer produces potential object locations in an image in the form of regions of interest. This method can ignore contextual information and procedurally generate bounding boxes based on the image size [6, 7], or use contextual information to generate bounding boxes [8, 9, 10, 11, 12]. The feature extractor creates high density information related to the features of the image. Classical machine learning methods are based on hand-crafted features such as gradients, color, etc [13, 14, 15, 16]. Deep learning object detectors typically make use of convolutional neural networks for feature extraction [11, 6, 7]. The classifier determines the class of the object based on the extracted features. Commonly used classical methods for classification include Bayes classification [17], k-nearest-neighbors [18], support vector machines [19], and decision trees [20]. Fully-connected neural networks have gained popularity recently [21, 22] and most state-of-the-art methods [23, 11] employ them for classification. Object detectors can be further divided between region proposal detectors, and regression detectors [21, 22].

Region proposal detectors have some method that creates regions of interest to be classified. This method can use classical computer vision [9] such as graph based image segmentation [8, 24] or a deep learning method such as a Region Proposal Network (RPN) [11, 12]. Feature extraction and classification is then performed for each region to determine the class of the object captured by the region. Regression methods overlay a regular grid on the image rather than having a distinct region proposal step. Feature extraction is performed on the entire image, and each grid space is classified [6, 7]. For both types of classifiers, bounding box regression is performed to improve localization performance, and Non-Maximum Suppression (NMS) is performed to remove redundant classifications [9, 10, 11, 6, 7].

3D Object Detection. Autonomous driving is a common application of 3D object detection [25, 26, 27], however this application is distinct from ours. Often times these systems rely on a LiDAR for point cloud generation. Due to the weight constraints of an aerial vehicle, our system does not implement a LiDAR. Additionally, the image capture locations in these systems are positioned along a parametric line along which the data was collected. These systems are able to process the input images similarly to a video stream, and propagate information between sequential frames. In our application, image capture locations are positioned along a parametric surface that follows the inspection surface rather than a one dimensional parametric line.

The most similar method to ours available is [28], but their method is intended for traffic sign detection. Their method was again intended for frames captured along a parametric line, but could be applied to a parametric surface with slight modifications. The main improvement of our method over [28], is the implementation of more powerful algorithms for point cloud generation and object detection.

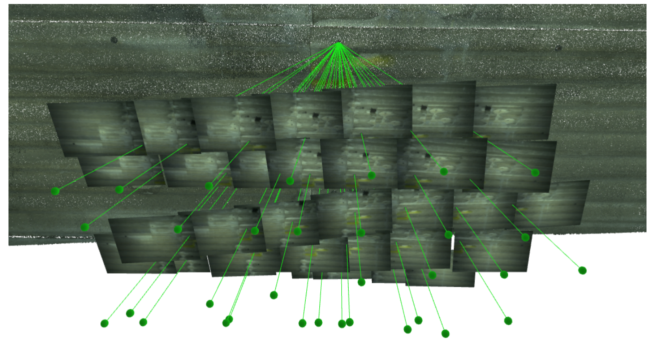


Figure 3: Tracing Pixel Locations of an Object to the 3D Location

3. Method

Our method consists of three stages as shown in Figure 2. The photogrammetry stage uses the set of input images to generate a 3D point cloud. The point cloud consists of keypoints which map a 3D point to the 2D pixel points in the images in which the 3D point is visible. The computer vision stage uses the set of input images and generates object detections in the form of image level bounding boxes with an associated class. The 3D localization stage uses the object detections and the keypoints to localize the objects in 3D space.

3.1 Photogrammetry

Pix4D [29] was used for the photogrammetry purposes of our method. Pix4D is able to process a set of images with no prior knowledge of the camera positions. This is highly desirable as the recorded position of the camera when the image is captured is subject to noise and drift, or may be completely unavailable in the worst cases. Pix4D generated an average of almost 3 thousand 3D keypoints per image for our dataset, and each of the keypoints was visible in an average of 4 images.

3.2 Computer Vision

Object detection was the selected computer vision task to localize the pixel locations of the defects in the images. Our method uses a deep learning method for object detection due to the significant increase in performance of deep learning methods observed in recent years [21, 22]. The current state-of-the-art methods use convolutional neural networks, and can be divided between two main groups, the region proposal methods and regression methods. Region proposal methods are typically slower than regression methods due to the evaluation of each region individually. However, region proposal methods typically have better performance than regression methods, especially on small objects [30]. Due to the nature of facility inspections, almost all of the objects are considered to be small to medium sized since they are smaller than 128 x 128 pixels. A plot of the pixel size of all object instances is shown in Figure 4. As such, the implemented architecture for our method was based on the R-CNN family of detectors [9, 10, 11, 12].

Our computer vision method uses the object detection portion of a ResNet18 Mask R-CNN [12] with slight modifications to the priors and training scheme. We did not implement the mask branch of Mask R-CNN, and uses only the class and bounding box outputs. Due to the differences in dataset, we made modifications to the anchor box priors, type of data augmentation, training parameters, and implemented class balancing techniques.

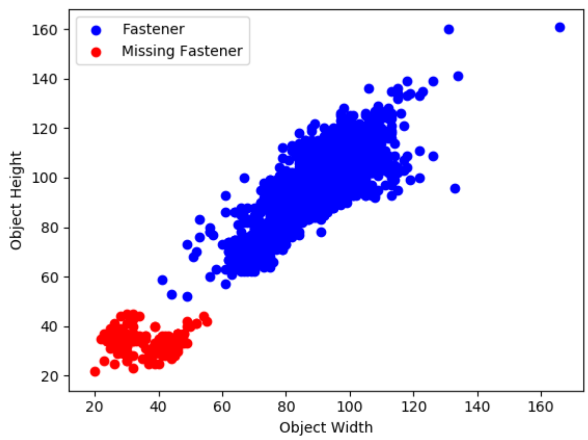


Figure 4: Image Object Instance Pixel Size

The anchor box sizes and aspect ratios used in the original Mask R-CNN implementation, and most object detectors, are rather large for our dataset. Our anchor box priors were selected using k-means clustering similarly to YOLOv3 [7]. We initialized two clusters per class, and maximized the average Intersection over Union (IoU) across all samples using k-means clustering.

During training the only form of data augmentation used by Mask R-CNN was horizontal flipping. This is likely because other forms of data augmentation would produce unrealistic images. Our dataset is composed of facility surfaces, which can be augmented more aggressively and still produce realistic images. As such, we implemented random horizontal flipping with probability of 50%, and random rotation in 90 degree increments, each with 25% likelihood.

Mask R-CNN used 16 images per batch during training, but in our experiments we used 1 image per batch as done in Faster R-CNN [11]. Additionally, Mask R-CNN uses 0.7 as the IoU threshold to classify an anchor box as a positive sample and an RPN batch size of 256 samples per image. For the classifier, Mask R-CNN assumes a positive to negative ratio between 1:3 and 1:1. Unlike the datasets used in object detection challenges [31, 32], our dataset is not object-centric and so with those parameters our positive to negative ratio was much lower. To increase the number of positive samples for the RPN, we set the objectness IoU threshold to 0.5. Even so, our dataset produces a low concentration of positive samples so as to better preserve the positive to negative ratio, we reduced the RPN batch sizes to 128 per image. For the classifier, we allow the positive to negative ratio to vary from 1:7 up to 1:1 by varying the classifier per image batch size and capping it at a maximum of 64 samples per image. However, in practice the ratio rarely gets above 1:3 for the classifier. Finally, since our dataset is significantly smaller than the that of object detection challenges, our training schedule was shorter than that of Mask R-CNN. We trained with a learning rate of 0.001 for 30k iterations, as this was the point that performance degraded on the validation data.

Our dataset featured severe class imbalance between the fasteners and missing fasteners. The number of missing fastener samples was about one eighteenth that of the number of fastener samples, and so we oversampled the images with missing fasteners eighteen times so that the same number of training iterations performed on fasteners and missing fasteners was equal. Additionally, since the fasteners were much larger than the missing fasteners, the fasteners had roughly five times the number of anchor boxes above the objectness IoU threshold per instance than missing fasteners. To

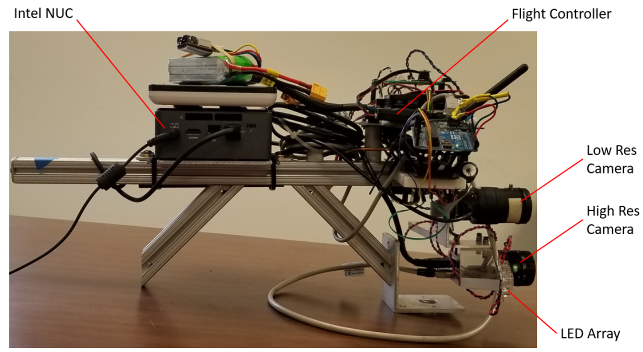


Figure 5: Handheld Data Collection System

combat this, during training we weighed the missing fasteners to be roughly five times as important as fasteners. Both of these efforts improved the performance of the model on missing fasteners.

3.3 3D Localization

Once the point cloud and pixel level object locations are created, the objects can be localized in 3D space. At first, the 2D pixel location of each keypoint is considered. A keypoint is associated with an object detection if it lies within the detection's bounding box. Each keypoint is associated with several images, and when two or more object detections share the same keypoint, the object detections are grouped together into an object set. Once all keypoints have been associated with object detections, and the all object detections grouped into object sets, the 3D location of the keypoints is considered. The keypoints associated with an object set are then clustered with mean shift clustering to determine an average 3D location of the keypoints. Mean shift was selected as it finds the maxima of the spatial density. A maxima in spatial density indicates that many keypoints are 'confident' that the object is located in that position. Figure 3 shows an example of the trace lines from the camera position, through the image plane at the position of the fastener, to the 3D location of a fastener.

4. Experiment

4.1 Dataset

The dataset collected was from a section of the interior NFAC walls over a 20 meters by 2 meters area using the handheld system shown in Figure 5. The images were collected at a distance of 1 meter from the wall, and captures on a grid-like pattern along the wall. The grid was spaced such that each object appears in 6-7 images per row, and in 4-5 rows of images. Objects along the border of the inspection area will appear in fewer images per row and/or fewer rows of images. The image capture locations were supervised by a Simultaneous Localization And Mapping (SLAM) algorithm to ensure that the spacing between capture locations, and the distance between the cameras to the wall was within tolerance.

Two types of labels were created for the dataset. First, the image object labels were created in the PASCAL VOC [31] format. Fasteners and missing fasteners instances were given a bounding box and class label. An example of objects from our dataset is shown in Figure 6. The second type of label created was the object set label. This label was the set of all image bounding boxes associated with the real world object.

Objects in the training, validation, and testing partitions must be distinct. Each real world object cannot appear on more than

Figure 6: Object Examples with Highlighted Fastener (Blue) and Missing Fastener (Red)

one of these partitions, as it would cause data leakage and unfairly increase the performance of the model. To ensure that each object appears in no more than one partition, the object sets were divided between the training, validation, and testing partitions. Objects along the border of the inspection area were not places in the validation or testing set as the reduced number of image instances is not representative of a full scale facility inspection. The object sets were divided in an 64/16/20 split between the training, validation, and test partitions respectfully. Images containing multiple objects were cropped into subimages such that each subimage contains objects of only one partition.

The training set contained 110 fastener object sets and 4 missing fastener object sets. Many of the training object sets were on the border of the inspection and so they did not contain the full 24 to 35 images per object set. The validation set contained 15 fastener object sets and 1 missing fastener object set. The test set contained 21 fastener object sets and 1 missing fastener object set. The object sets in both the validation and test set contained the full 24 to 35 images per object set.

4.2 Results

No analysis of the accuracy of the photogrammetry was performed as Pix4D provides its own analysis. We use frame-mAP and video-mAP as defined by [33] to quantitatively evaluate our results. Frame-mAP is used to evaluate the object detector and measures the mean Average Precision (mAP) on the image level labels. For video-mAP, we use the ground truth object set labels in place of actual video labels. Video-mAP measures the mAP of the full method per object set. For both metrics we use an IoU threshold of 0.5 to be considered a true positive. The results are shown in Table 1.

On the per frame level, our method had an average precision of 92.1% when detecting fasteners, and 31.0% when detecting the missing fasteners in the test set. On the per object level, the object detector had an average precision of 86.4% when detecting fastener object sets and 0.00% when detecting missing fastener object sets. For the single missing fastener object set in the test set, 37% of the image instances were properly detected. Unfortunately since this is less than 50% of the image instances, it was not considered an object set detection for the video-AP. The decrease in performance on the missing fasteners is likely due to the both their smaller size and the smaller number of samples to train upon.

Unfortunately, due to the relatively small amount of object sets, the results are not conclusive. The results show potential as we were able to locate most of the fastener objects on both the image level and in 3D space. However, in the case of missing fasteners there was only one object set to test upon, and so the results are unreliable.

Class	Frame-AP	Video-AP
Fastener	92.1	86.4
Missing Fastener	31.0	0.00
Mean	61.5	43.2

Table 1: mAP Performance

5. Conclusion

Our method performed well and our results are promising. This method has the potential to enable quick and inexpensive inspection of large wind tunnel facilities. We were able to successful perform object detection in the 2D raw images and localize the defects in 3D space. Additionally, the inference time was on the order of seconds per image using a single Tesla V100 GPU. With a reasonable number of GPUs the entire facility can be processed over a week. More importantly, this process of utilizing drones requires significantly less tunnel downtime than with manual inspections, and will have easier access to the difficult to reach areas of the tunnel. We intend to continue this project and collect significantly more data to improve the performance of the model, and to properly test and evaluate this method.

6. Acknowledgements

We are grateful for the support from Fluid Mechanics Lab at Ames Research Center for sponsoring this project, AEDC for allowing us access to the NFAC facility, and the NASA Advanced Supercomputing (NAS) Division at Ames Research Center for supporting us with the required computational resources.

References

[1] J. Dino, "NASA - National Full-Scale Aerodynamics Complex (NFAC)."

[2] David W. James, Juergen Eckermann, Fawsi Belblidia, and Johann Sienz, "Point cloud data from Photogrammetry techniques to generate 3D Geometry," *Proceedings of the 23rd UK Conference of the Association for Computational Mechanics in Engineering 8*, no. April, p. 5, 2015.

[3] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision*. Cambridge university press, 2003.

[4] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop," 2015.

[5] S. Vijayanarasimhan and K. Grauman, "What's it going to cost you?: Predicting effort vs. informativeness for multi-label image annotations," *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*, vol. 2009 IEEE, no. June, pp. 2262–2269, 2009.

[6] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg, "Ssd: Single shot multibox detector," in *European conference on computer vision*, pp. 21–37, 2016.

[7] J. Redmon and A. Farhadi, "YOLOv3: An Incremental Improvement," 2018.

[8] J. R. R. Uijlings, K. E. A. Van De Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International journal of computer vision*, vol. 104, no. 2, pp. 154–171, 2013.

[9] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 580–587, 2014.

[10] R. Girshick, "Fast R-CNN," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2015 Inter, pp. 1440–1448, 2015.

[11] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in neural information processing systems*, pp. 91–99, 2015.

[12] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask r-cnn," in *arXiv preprint arXiv:1703.06870*, pp. 2961–2969, 2017.

[13] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European conference on computer vision*, pp. 404–417, 2006.

[15] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "ORB: An efficient alternative to SIFT or SURF.," in *ICCV*, vol. 11, p. 2, 2011.

[16] M. Brown, R. Szeliski, and S. Winder, "Multi-image matching using multi-scale oriented patches," in *CVPR*, pp. 510–517, 2005.

[17] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[18] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 4, pp. 325–327, 1976.

[19] C. Cortes and V. Vapnik, "Support-vector networks," *Machine learning*, vol. 20, no. 3, pp. 273–297, 1995.

[20] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and regression trees*. Routledge, 2017.

[21] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object Detection With Deep Learning: A Review," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2019.

[22] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and Others, "Speed/accuracy trade-offs for modern convolutional object detectors," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 7310–7311, 2017.

[23] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 779–788, 2016.

[24] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[25] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3D Object Detection for Autonomous Driving," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 2147–2156, 2016.

[26] C. Xiaozhi, M. Huimin, W. Ji, L. Bo, and X. Tian, "Multi-View 3D Object Detection Network for Autonomous Driving," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1907–1915, 2017.

[27] Y. Zhou and O. Tuzel, "VoxelNet: End-to-End Learning for Point Cloud Based 3D Object Detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4490–4499, 2018.

[28] V. Balali and M. Golparvar-Fard, "Recognition and 3D Localization of Traffic Signs via Image-Based Point Cloud Models," *Congress on Computing in Civil Engineering, Proceedings*, pp. 206–214, 2015.

[29] "Professional photogrammetry and drone mapping software | Pix4D,"

[30] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[31] M. Everingham, L. Van˜Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results." http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html.

[32] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*, pp. 740–755, Springer, 2014.

[33] G. Gkioxari and J. Malik, "Finding action tubes," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 759–768, 2015.

A Review of Rapidly Exploring Random Tree Path Planning Algorithms

Jacqueline Jermyn, Rodney Roberts

FAMU-FSU College of Engineering
2525 Pottsdamer Street
Tallahassee, Florida, 32310

jmj15c@my.fsu.edu, rroberts@eng.famu.fsu.edu

ABSTRACT

Path planning algorithms are robot navigation techniques that determine a robot's route from its beginning to its goal configurations. The Rapidly Exploring Random Tree (RRT) algorithm is a sampling-based path planning algorithm, which begins at an initial configuration and builds a tree containing vertices and edges until it reaches its goal configuration. It is extensively implemented because of its simplicity, but it does not determine optimal paths. As a result, various modifications have been made to improve its performance. This paper reviews simulations to evaluate the performance of the RRT, Rapidly Exploring Random Tree Star (RRT*), and Informed RRT* algorithms.

Keywords

Informed Rapidly Exploring Random Tree Star (Informed RRT*), Path Planning, Rapidly Exploring Random Tree (RRT), Rapidly Exploring Random Tree Star (RRT*)

1. INTRODUCTION

In a robotic navigation scenario, a robot's environment is also known as the configuration space C to include free space C_{free} and obstacle space C_{obs}. Free space encompasses all spaces that the robot can occupy. Obstacle space contains obstructions that restrict a robot's movements. Consequently, free space and obstacle space are complements of each other. Path planning algorithms determine routes to move a robot from a start configuration $\mathbf{q}_0$ to a goal configuration $\mathbf{q}_G$ in free space C_{free} without colliding with the obstacle space C_{obs} [16]. Figure 1 depicts a generalized path planning environment showing free space, obstacle space, the initial configuration, and the goal configuration.

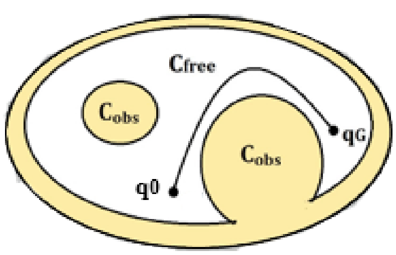


Figure 1. Generalized Path Planning Environment

Path-planning techniques are divided into combinatorial and sampling-based planning methods [22]. Combinatorial methods, such as trapezoidal decomposition, build structures in the configuration space that discretely and completely obtain all the necessary data to plan a path. [16]. These algorithms are complete because they either find a path or they determine that a path does not exist [15]. In addition, these methods determine an exact representation of the free space, build a roadmap, which is a graph that characterizes free space, and generates a path using this road map. Combinatorial planning algorithms, such as triangular decomposition, may have a fast convergence time, but they are impractical to implement because of their high complexity associated with dividing the free space into triangles [15].

Sampling-based methods employ collision detection to repeatedly search the configuration space for a path instead of characterizing the entire configuration space [16]. Sampling-based methods, such as RRT, have longer convergence times than combinatorial methods [15]. However, sampling-based methods are commonly used for industrial grade problems because they are simple to implement [14], [23]. In addition, sampling-based methods will generate a path, if it exists, but the runtime of the algorithm may be infinite [1]. Figure 2 shows the classification of path planning algorithms.

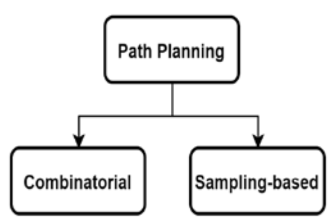


Figure 2. Classification of Path Planning Algorithms

The remainder of this paper is organized as follows: section 2 will review three types of rapidly exploring random tree algorithms to include the RRT, RRT*, and Informed RRT* algorithms. Section 3 will discuss simulations to evaluate their performance in two simulated environments, which are a L-shaped wall of obstacles and a wall of obstacles that has gaps in it. Section 4 will conclude with the highlights of these simulations.

2. RAPIDLY EXPLORING RANDOM TREE (RRT) ALGORITHMS

The Rapidly Exploring Random Tree (RRT) algorithm efficiently generates paths, while avoiding obstacles [14], [10]. Variants of this algorithm that will be discussed in this paper include the RRT, RRT*, and Informed RRT* algorithms.

2.1 Rapidly Exploring Random Tree (RRT)

The RRT algorithm creates a path from an initial configuration $\mathbf{q}_0$ to a goal configuration $\mathbf{q}_{goal}$. It then constructs a tree G, which consists of vertices V and edges E. The start location $\mathbf{q}_0$ and goal location $\mathbf{q}_{goal}$ are automatically added to the vertices list.

The tree begins at $\mathbf{q}_0$. It then randomly selects a point $\mathbf{q}_{rand}$ in the configuration space. If this point coincides with an obstacle, the algorithm selects a different random point. If the randomly generated point $\mathbf{q}_{rand}$ is in the free space, the algorithm determines the nearest point in the tree $\mathbf{q}_{nearest}$ to $\mathbf{q}_{rand}$.

This algorithm then attempts to draw a line from its nearest point $\mathbf{q}_{nearest}$ to the randomly generated point $\mathbf{q}_{rand}$. If this line collides with an obstacle, the algorithm selects a new random point. If the line does not collide with an obstacle, this line is drawn between the nearest point in the tree and the randomly generated point. The algorithm terminates when the tree arrives at the goal location $\mathbf{q}_{goal}$ [16]. This algorithm outputs tree G and Q_{sol}, which is the path from the start to the goal positions [14]. Table 1 contains the pseudocode for the RRT.

Table 1. RRT Algorithm Pseudocode

```
RRT(q_0, q_goal)
  V ← {q_0, q_goal}
  E ← ∅
  G = (V, E)
  Q_sol ← ∅
  while GOAL_REACHED(q_rand, q_goal) == false
      q_rand ← RANDOM_CONFIG()
      if OBSTACLE_FREE(q_rand) == true
          q_nearest ← NEAREST(G, q_rand)
          if COLLISION_FREE(q_nearest, q_rand) == true
               G ← INSERT_NODE(q_nearest, q_rand, G)
               G ← EXTEND(q_near, q_rand)
          end
      end
  end
   Q_sol ← LIST_OF_COMPLETED PATHS(G)
  return G, Q_sol
end
```

2.2 RRT* Algorithm

The Rapidly Exploring Random Tree Star (RRT*) algorithm is an optimized RRT algorithm that incorporates near neighbor search and rewiring tree functions to reduce the distance traveled [17]. The RRT* algorithm has the asymptotic optimality property; consequently, it converges to an optimal path, although it has a long convergence time [8], [11], [12].

This algorithm begins by initializing the search tree by adding the start configuration $\mathbf{q}_0$ and the goal configuration $\mathbf{q}_{goal}$ to the vertices list. It generates a random configuration $\mathbf{q}_{rand}$ in the free space, and the algorithm determines the nearest point $\mathbf{q}_{nearest}$ in the tree to $\mathbf{q}_{rand}$. If the path from $\mathbf{q}_{neares}$ to $\mathbf{q}_{rand}$ is obstacle free, the near neighbor search function looks for nodes that are within a radius k that is selected by the researcher. The choose parent function then selects the optimal configuration from the nearby configurations $\mathbf{q}_{min}$, and it is added to tree G. The rewire function compares the distance traveled through $\mathbf{q}_{min}$ to the original distance to minimize the distance traveled [17]. This algorithm terminates when the number of searches N is reached. It then outputs tree G and Q_{sol} which is the path from the start to the goal positions. Table 2 contains the pseudocode for the RRT* algorithm.

Table 2. RRT* Algorithm Pseudocode

```
RRT*(q_0, q_goal, N, k)
  V ← {q_0, q_goal}
  E ← ∅
  G = (V, E)
  Q_sol ← ∅
  for i = 0 to N
      q_rand ← RANDOM_CONFIG()
      q_nearest ← NEAREST(G, q_rand)
        if OBSTACLE_FREE(q_rand, q_nearest)
           q_near ← NEAR(G, q_rand, k)
           q_min ← CHOOSE_PARENT(q_near, q_nearest, q_rand)
           G ← INSERT_NODE(q_min, q_rand, G)
           G ← REWIRE(G, q_near, q_min, q_rand)
        end
  end
  Q_sol ← LIST_OF_COMPLETED_PATHS(G)
  return G, Q_sol
end
```

2.3 Informed RRT* Algorithm

The Informed RRT* algorithm improves upon the RRT* algorithm by reducing the size of the search area, which results in shorter paths [3], [4]. The RRT and RRT* algorithms sample the entire search area, which cause their search trees to include nodes that are far away from the start and goal configurations. Paths that include nodes that are far away cause robots to travel long distances to get to their destinations. Instead, the Informed RRT* algorithm is most concerned with the area in close proximity to the start and goal locations. It samples the elliptical subset of the search area that has the start and goal locations as this ellipse's foci [5], [6]. Fig. 3 shows a diagram of this search ellipse that illustrates that c_{min} is the minimum distance from $\mathbf{q}_0$ to $\mathbf{q}_{goal}$. Table 3 contains the pseudocode for the Informed RRT* algorithm.

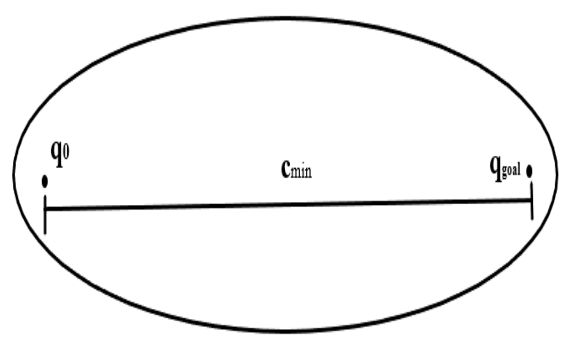


Figure 3. Search Ellipse for Informed RRT* Algorithm

Table 3. Informed RRT* Algorithm Pseudocode

```
Informed RRT*(q_0,q_goal)
    V ← {q_0, q_goal}
    E ← ∅
    G = (V, E)
    Q_sol ← ∅
    for i = 0 to N
        if Q_sol == ∅
            G ← RRT*(q_0, N)
            Q_sol ← LIST_OF_COMPLETE_PATHS(G)
        else
            c_min ← MIN(COST(q|q ∈ Q_sol))
            q_rand ← SAMPLE(q_0, q_goal, c_min)
            G ← RRT*(G, q_rand)
        end
    Q_sol ← LIST_OF_COMPLETED_PATHS(G)
    return G, Q_sol
end
```

3. SIMULATIONS

Two-Dimensional (2D) simulations for the RRT, RRT*, and Informed RRT* algorithms were conducted using the PythonRobotics toolbox [20], [21], [22].

Each algorithm's Python file was opened to initialize the experimental parameters. The maximum number of searches for each variation of RRT was set to 200. The boundaries of the search area were initialized by setting the maximum and minimum values of the x and y axes to -2 and 15 respectively. The start and goal nodes of the search trees were set to (0,0) and (10,10) respectively. In addition, two circular obstacle lists, as shown in Tables 4 and 5, were created so that the performance of each algorithm could be tested in two simulated environments. The first set of obstacles, shown in Figure 4, form an L shaped wall, while the second set of obstacles, shown in Figure 5, form a wall with narrow gaps between the obstacles.

Table 4. Obstacle List 1

X Coordinate of Center	Y Coordinate of Center	Radius
5	5	1
3	6	2
3	10	2
3	10	2
7	5	2
9	5	2
8	10	1
6	12	1

Table 5. Obstacle List 2

X Coordinate of Center	Y Coordinate of Center	Radius
5	5	0.5
9	6	1
7	5	1
1	5	1
3	6	1
7	9	1

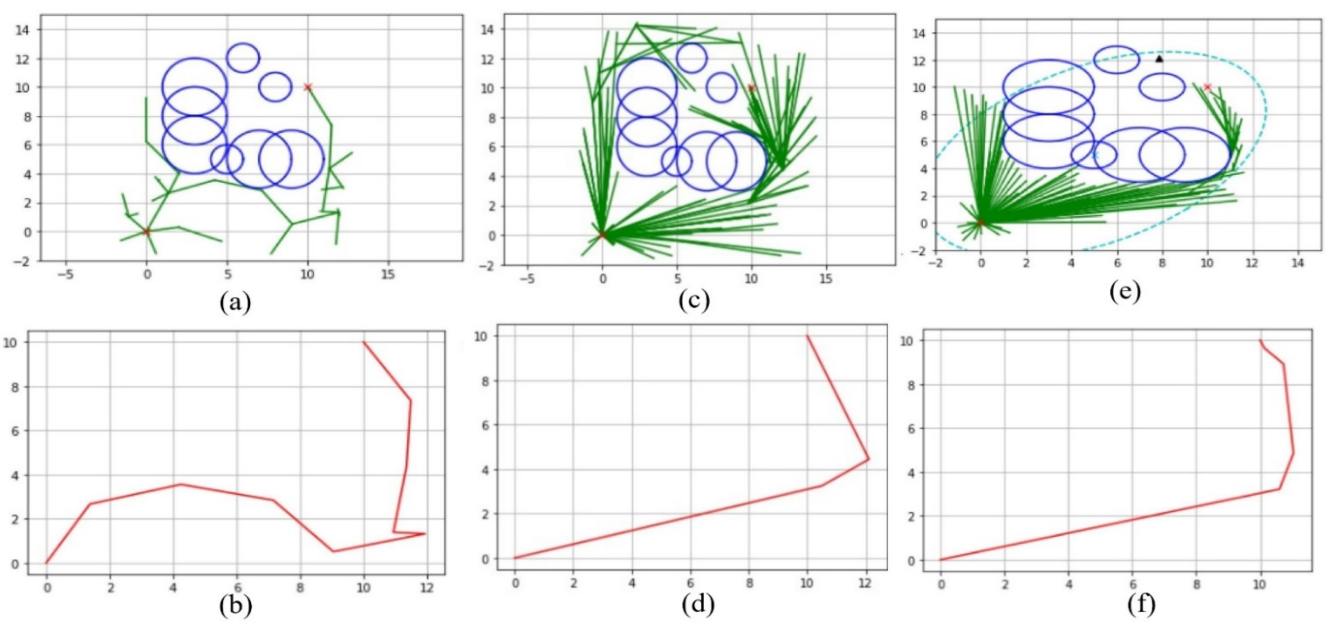


Figure 4. Simulations for Obstacle Set 1. Figures (a) and (b) correspond to Trial 2 for the RRT algorithm simulation. There are 11 nodes in path (b) because the second to the last node is located at (10.01, 9.97). The length of path (b) is 25.04. Figures (c) and (d) correspond to Trial 9 for the RRT* algorithm simulation. There are 5 nodes in path (d) because the second to the last node is located at (11.04, 7.02). The length of path (d) is 18.94. Figures (e) and (f) correspond to Trial 8 for the Informed RRT* algorithm. Figure (e) shows that the foci of the search ellipse are located at the start and goal locations. There are 6 nodes in path (f) because the second to the last node in this path is located at (10.13, 9.65), and the length of this path is 18.18.

The runtime, number of searches, and total number of nodes in the search tree were recorded for each simulation. If an algorithm successfully generated a path, the number of nodes in the path and the coordinates of nodes in this search path were also recorded. The data for these simulations were recorded in an Excel spreadsheet.

For these simulations, the distance between each set of consecutive points was calculated as

$$d_{i,i+1} = \sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2} \quad (1)$$

where x_i and y_i are the coordinates of the current point, and x_{i+1} and y_{i+1} are the coordinates of the next point in the path. The total length of each search path d_{tot} was computed as

$$d_{tot} = \sum_{i=1}^{n} d_{i,i+1} \quad (2)$$

where n is the total number of points in the path, and $d_{i,i+1}$ is the distance between each pair of consecutive points.

Each simulation was repeated for at least 30 trials or until a path could be found. The Central Limit Theorem states that the distribution of a sample approaches a Gaussian distribution as the number of samples increases [7].

The sample statistics for the runtime, number of searches, total number of nodes in the search tree, number of nodes in the path, and total path length were calculated and recorded in Tables 6, 7, 8, 9, 10, and 11 in the appendix. The runtime includes the amount of time for each variation of RRT algorithm to build a search tree and to generate the path from the start to goal locations. The total number of searches is equal to the total number of iterations for each algorithm. The number of searches for the RRT algorithm is less than 200 because this algorithm terminates as soon as a path from the start to the goal node is generated. By contrast, the number of searches for the RRT* and Informed RRT* algorithms are equal to 200 because these algorithms continue to improve the search path until the maximum number of searches is reached.

The number of nodes in the search tree include only the number of points that are successfully added to the search tree. Randomly selected points that are generated inside of the obstacle space are not added to the search tree. In addition, randomly generated points that are blocked by obstacles are not added to search trees. Furthermore, the number of nodes in the path are equal to number of points in the path from the start position to the goal position.

Sample statistics, to include the mean, median, mode, and standard deviation, are calculated to compare the performance of each algorithm [8].

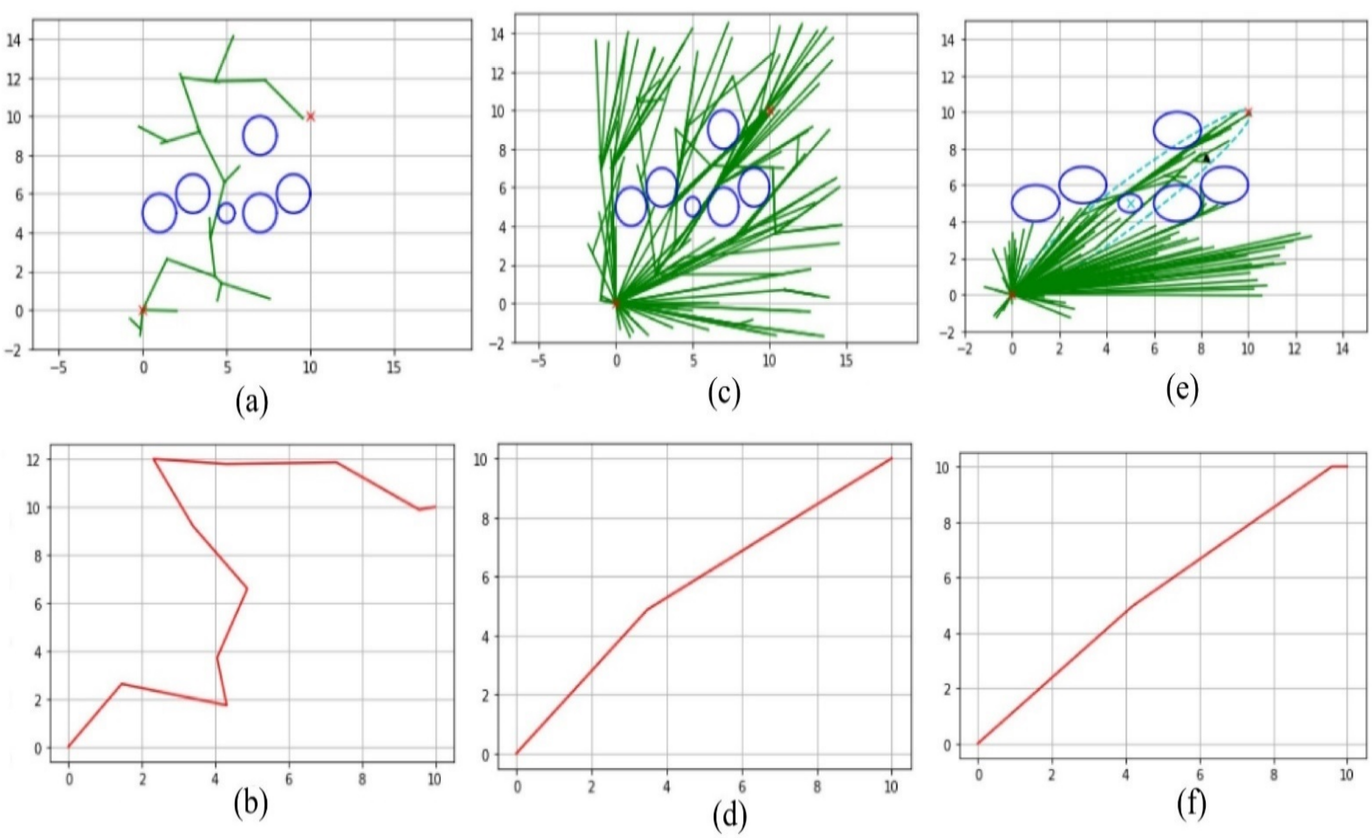


Figure 5. Simulations for Obstacle Set 2. Figures (a) and (b) correspond to Trial 5 for the RRT algorithm simulation. There are 11 nodes in path (b) because the second to the last node is located at (9.56, 9.88). The length of path (b) is 19.44. Figures (c) and (d) correspond to Trial 2 for the RRT* algorithm simulation. There are 5 nodes in path (d) because there are two nodes located at (9.31, 7.02) and (6.43, 5.91). The length of path (d) is 14.69. Figures (e) and (f) correspond to Trial 27 for the Informed RRT* algorithm. Figure (e) shows that the foci of the search ellipse are located at the start and goal locations. There are 4 nodes in path (f) because the second to the last node in this path is located at (9.58, 9.99), and the length of this path is 14.29.

The arithmetic mean is the average value of a quantity [8], [10]. The mean $\bar{x}$ is given as

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \quad (3)$$

where x_i is a data value, and n is the total number of datapoints. The median is the middle value of the data set, the mode is of a set of datapoints is the value that occurs most often, and standard deviation is an indication of how far datapoints are spread apart [8] [10].

The standard deviation *s* is computed as

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x-\bar{x})^2} \quad (4)$$

where *n* is the total number of samples, *x* is an individual sample, and $\bar{x}$ is the mean.

The data for the first obstacle set is provided in Tables 6, 7, and 8, which are found in the appendix section. The runtimes for the RRT, RRT*, and Informed RRT* simulations corresponding to obstacle set 1 were 0.51s, 0.47s, and 0.54s with standard deviations of 0.26s, 0.25s, and 0.31s respectively. The mean number of searches for the RRT algorithms was 42.6 searches with a standard deviation of 25.88. The RRT* and Informed RRT* algorithm each made 200 searches. The mean pathlengths for the RRT, RRT*, and Informed RRT* algorithms were 23.76, 19.065, and 18.185 with corresponding standard deviations of 2.93, 0.57, and 0.25 respectively. Consequently, the Informed RRT* algorithm found the shortest path lengths, and its path length is the most predictable because its standard deviation is the smallest. The RRT and RRT* algorithms successfully found a path from the start to goal locations for every trial; consequently, these simulations were run 30 times each. The Informed RRT* algorithm failed to find a path for some of the trials; consequently, it was simulated for 72 trials so that 30 path length measurements could be obtained. The Informed RRT* algorithm found a path for only 41.67% of the trials.

The data for the second obstacle set is provided in Tables 9, 10, and 11 in the appendix section. The runtimes for the RRT, RRT*, and Informed RRT* simulations corresponding to obstacle set 2 were 0.53s, 0.45s, and 0.47s with standard deviations of 0.30s, 0.23s, and 0.24s respectively. The mean number of searches for the RRT algorithm was 20.87 searches with standard deviations of 9.48. The RRT* and Informed RRT* algorithm each made 200 searches. The mean pathlengths for the RRT, RRT*, and Informed RRT* algorithms were 19.60, 14.59, and 14.28 with corresponding standard deviations of 2.93, 0.37, and 0.15 respectively. Consequently, the Informed RRT* algorithm found the shortest path lengths, and its path length was the most predictable because its standard deviation is the smallest. The RRT and RRT* algorithms successfully found a path from the start to goal locations for every trial; consequently, these simulations were run 30 times each. The Informed RRT* algorithm failed to find a path for some of the trials; consequently, it was simulated for 32 trials so that 30 path length measurements could be obtained. The Informed RRT* algorithm found a path for 94% of the trials.

The mean path lengths for obstacle set 1 were longer than the path lengths for obstacle set 2 because the path planning algorithms traveled through gaps in the obstacles, as illustrated in Figure 5. For the Informed RRT* algorithm, the success rate for obstacle set 1 was lower than the success rate for obstacle set 2. For obstacle set 1, the algorithm had difficulty finding a path around the L-shaped block of obstacles because it focused on the search area between the start and goal locations.

4. CONCLUSIONS AND FUTURE WORK

Path planning algorithms are sampling-based robot navigation techniques that determine a robot's route from its beginning to its goal configurations. This paper reviewed three variations of the RRT algorithms to include the RRT, RRT*, and Informed RRT* algorithms. Simulations were also conducted to compare their performances. All three algorithms were able to reach the goal locations. The RRT algorithm had the longest path lengths, and the Informed RRT* algorithm had the shortest path lengths. The success rates for the RRT and RRT* algorithms were 100% for both obstacle sets. However, the success rate of the Informed RRT* was 41.67% for obstacle set 1 and 94% for obstacle set 2. As a result, the effectiveness of this path planning algorithm depends on the arrangements of obstacles in the environment.

Future experiments could include changing the sampling method for the Informed RRT* algorithm to improve its success rate. Also, the RRT, RRT*, and Informed RRT* algorithms could be modified to sample three-dimensional areas so that these algorithms could be applied to unmanned aerial vehicles and unmanned underwater vehicles, which travel in three dimensions.

5. ACKNOWLEDGMENTS

This research was funded by the FAMU-FSU College of Engineering Dean's Fellows Program.

6. REFERENCES

[1] M. Elbanhawi and M. Simic, 2014. Sampling-Based Robot Motion Planning: A Review. *IEEE Access* 2 (February 2014), 56–77. DOI:http://dx.doi.org/10.1109/access.2014.2302442

[2] J. D. Gammell. 2014. Informed RRT*. (March 2014). Retrieved March 2, 2020 from https://robotic-esp.com/code/informed-rrtstar/

[3] J. D. Gammell. 2014. Informed RRT* @ UTIAS (IROS 2014). (July 2014). Retrieved February 20, 2020 from https://www.youtube.com/watch?v=nsl-5MZfwu4

[4] J. D. Gammell, Timothy D. Barfoot, and Siddhartha S. Srinivasa. 2018. Informed Sampling for Asymptotically Optimal Path Planning. *IEEE Transactions on Robotics* 34, 4 (2018),966–984. DOI:http://dx.doi.org/10.1109/tro.2018.2830331

[5] J. D. Gammell, Siddhartha S. Srinivasa, and Timothy D. Barfoot. 2014. Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems* (2014), 2997–3004. DOI:http://dx.doi.org/10.1109/iros.2014.6942976

[6] A. Ganti. 2020. What Is the Central Limit Theorem (CLT)? (January 2020). Retrieved February 26, 2020 from https://www.investopedia.com/terms/c/central_limit_theorem.asp

[7] L. Green. 2008. Mean, Mode, Median, and Standard Deviation. (2008). Retrieved February 26, 2020 from http://www.ltcconline.net/greenl/courses/201/descstat/mean.htm

[8] B. C. Guevara, "An Overview of the Class of Rapidly-Exploring Random Trees," M.S. thesis, Dept. Humanities., Utrecht Univ., Utrecht, Netherlands, 2018.

[9] B. L. Illowsky and S. L. Dean. 2018. *Introductory Statistics*, Houston, TX: OpenStax, Rice University.

[10] F. Islam, J. Nasir, U. Malik, Y. Ayaz, and O. Hasan. 2012. RRT∗-Smart: Rapid Convergence Implementation of RRT∗ Towards Optimal Solution. *2012 IEEE International Conference on Mechatronics and Automation* (2012). DOI:http://dx.doi.org/10.1109/icma.2012.6284384

[11] S. Karaman, and E. Frazzoli, 2011. Sampling-based Algorithms for Optimal Motion Planning. *The International Journal of Robotics Research* 30, 7 (2011), 846–894. DOI:http://dx.doi.org/10.1177/0278364911406761

[12] S. Karaman, M. R. Walter, A. Perez, E. Frazzoli, and S. Teller, 2011. Anytime Motion Planning using the RRT*. *2011 IEEE International Conference on Robotics and Automation* (June 2011). DOI:http://dx.doi.org/10.1109/icra.2011.5980479

[13] Min-Cheol Kim and Jae-Bok Song. 2017. Informed RRT* with improved converging rate by adopting wrapping procedure. *Intelligent Service Robotics* 11, 1 (February 2017), 53–60. DOI:http://dx.doi.org/10.1007/s11370-017-0242-9

[14] S.M. LaValle. 1998. Rapidly-Exploring Random Trees: A New Tool for Path Planning. (October 1998).

[15] S. M. LaValle. 2006. Planning Algorithms. Cambridge University Press, New York, NY, USA.

[16] S. M. LaValle, 2011. Motion Planning: The Essentials. *IEEE Robotics & Automation Magazine* 18, 1 (March 2011), 79–89. DOI:http://dx.doi.org/10.1109/mra.2011.940276

[17] I. Noreen, A. Khan and Z. Habib. 2016. A Comparison of RRT, RRT*, and RRT*-Smart Path Planning Algorithms. *IJCSNS International Journal of Computer Science and Network Security* 16, 10 (October 2016). DOI:http://dx.doi.org/10.1109/icma.2012.6284384

[18] A. Sakai. 2020. AtsushiSakai/PythonRobotics. (February 2020). Retrieved February 20, 2020 from https://github.com/AtsushiSakai/PythonRobotics

[19] A. Sakai. 2018. Welcome to PythonRobotics's documentation! (2018). Retrieved February 20, 2020 from https://pythonrobotics.readthedocs.io/en/latest/index.html#

[20] A. Sakai, Daniel Ingram, Joseph Dinius, Karan Chawla, Antonin Raffin and Alexis Paques. (2018). PythonRobotics: a Python code collection of robotics algorithms. *ArXiv, abs/1808.10703*.

[21] O. Salzman. 2019. Sampling-Based Robot Motion Planning. Commun. ACM 62, 10 (September 2019), 54–63. DOI:https://doi-org.proxy.lib.fsu.edu/10.1145/3318164

[22] O. Soussi, R. Benatitallah, D. Duvivier, A. Artiba, N. Belanger, and P. Feyzeau, In 5th *IESM Conference*, (Rabat, MA), Institute of Electrical and Electronics Engineers.

7. APPENDIX

The data for the RRT, RRT* and Informed RRT* algorithms corresponding to obstacle list 1 are provided in Tables 6, 7, and 8 respectively.

Table 6. RRT Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.51	0.55	0.06	0.95	N/A	0.26
Number of Searches	42.63	37	10	146	37	25.88
Total Number of Nodes in Tree	24.73	21	10	97	28	16.52
Number of Nodes in Path	9.97	10	8	13	10	1.19
Path Length	23.76	22.81	19.88	30.59	N/A	2.93

Table 7. RRT* Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.47	0.47	0.06	0.97	N/A	0.26
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	149	149	125	166	149	11.4078
Number of Nodes in Path	4.87	5	4	6	5	0.43
Path Length	19.07	18.98	18.09	20.73	N/A	0.57

Table 8. Informed RRT* Results for Obstacle List 1

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.54	0.60	0.02	1.00	N/A	0.31
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	111.99	112.50	80	150	100	15.95
Number of Nodes in Path	6.20	6	5	8	6	0.85
Path Length	18.19	18.09	17.86	18.82	N/A	0.25

The data for the RRT, RRT* and Informed RRT* algorithms corresponding to obstacle set 2 are provided in Tables 9, 10, and 11 respectively.

Table 9. RRT Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.53	0.58	0.003	0.94	N/A	0.30
Number of Searches	20.87	21	6	53	21	9.48
Total Number of Nodes in Tree	15.33	13	6	48	13	8.32
Number of Nodes in Path	8.2	8	6	12	8	1.49
Path Length	19.60	19.27	14.32	27.95	N/A	3.07

Table 10. RRT* Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.45	0.48	0.03	0.89	N/A	0.23
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	183.60	184	175	191	185	4.23
Number of Nodes in Path	4.23	4	4	5	4	0.43
Path Length	14.59	14.51	14.21	15.9	N/A	0.37

Table 11. Informed RRT* Results for Obstacle List 2

Quantity	Mean	Median	Min	Max	Mode	Sample Standard Deviation
Runtime	0.47	0.52	0.05	0.97	N/A	0.24
Number of Searches	200	200	200	200	200	0
Total Number of Nodes in Tree	171.75	176.50	133	187	178	13.38
Number of Nodes in Path	4.6	4.5	4	6	4	0.67
Path Length	14.28	14.24	14.19	14.98	N/A	0.15

Vision-based decision system for humanoid robot

Bartosz Bok
Warsaw University of Technology
Faculty of Electronics
and Information Technology
bok.bartosz@gmail.com

Dominik Górczyński
Warsaw University of Technology
Faculty of Power
and Aeronautical Engineering
gorczyn1892@gmail.com

Kacper Mikołajczyk
Warsaw University of Technology
Faculty of Power
and Aeronautical Engineering
kacper.mikolajczyk96@gmail.com

Maksymilian Szumowski
Warsaw University of Technology
Institute of Aeronautics
and Applied Mechanics
mszumowski@meil.pw.edu.pl

ABSTRACT

This article presents the method of using a vision system based on a camera, which is the main source of information about the surroundings of a humanoid robot. Basing on the image analysis, the robot is able to make a decision about its next moves. The decision system based on the camera image is shown on the example of sumo fighting humanoid robots, competition that become increasingly popular in the world. This article describes the use of Hough transform on the camera image, which allows to determine the position of the robot on the ring. Then, relying on the calculation of the largest gradient tangent to the line of the ring, robot can find the position of the opponent. Thanks to this, the decision making system is able to calculate the estimated trajectory based on the opponent's movement history, which allows the robot to approach opponent as soon as possible. The above-mentioned method has been implemented on real humanoid. The article contains the results of a robot test in the ring, which is used in competitions.

Keywords: vision system, decision system, humanoid robot.

1. Introduction

Technological development has recently changed the reality radically. People often use technology to make their lives easier. Robots will increasingly accompany them in private and professional life. They will have to move in a space where people also move, so there is a problem because many of them move thanks to the wheel drive. This is a good way only on a flat surface. However, the space in which people move is adapted for bipeds. It has a lot of flat surface, as well as many places that can be obstacles for robots, such as stairs, curbs or other uneven terrain. Therefore, a good solution may be the development of robots that move like humans, i.e. humanoids. Recently, the topic of humanoid robots is becoming more and more popular. The world of science is trying to popularize solutions of bipedal robots. There are many competitions in which such robots can compete with each other, such as races, matches or robot battles. Many robotics are involved in the development of algorithms for controlling robots [1], [2]. In order to make the right decisions, the robot needs to get information about its surroundings. For this purpose, most robots have different types of sensors, e.g. distances (ultrasonic, laser, etc.), color. However, when there is information about distance or color, robot do not get information about what object it is or where it can be in space. The LIDAR sensor gives information about the distances of objects on the plane around the robot, however also no major information about objects is obtained, e.g. what is the color distribution in space [3].

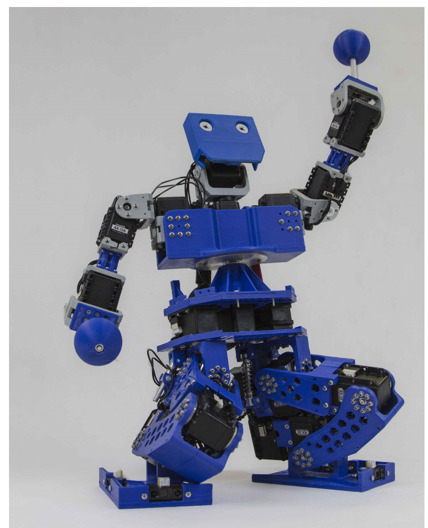


Figure 1. Melson, a humanoid robot designed and built by members of Student's Robotics Associacion at Warsaw University of Technology

By using the camera as the main source of information about the environment and appropriate analysis of its image, robot is able to know the characteristics of the space around the robot much more accurately. It is useful when planning the robot's paths of travel [4] or various types of humanoid robot jobs [5].

2. Humanoid Robot Melson

In the Students' Robot Association at the Warsaw University of Technology, a humanoid robot has been prepared with a developed autonomous operation. The robot is called 'Melson', which has a height of 50 cm and a weight of 2.6 kg. He has 19 degrees of freedom, distance sensors and a camera on his head. The robot is au-

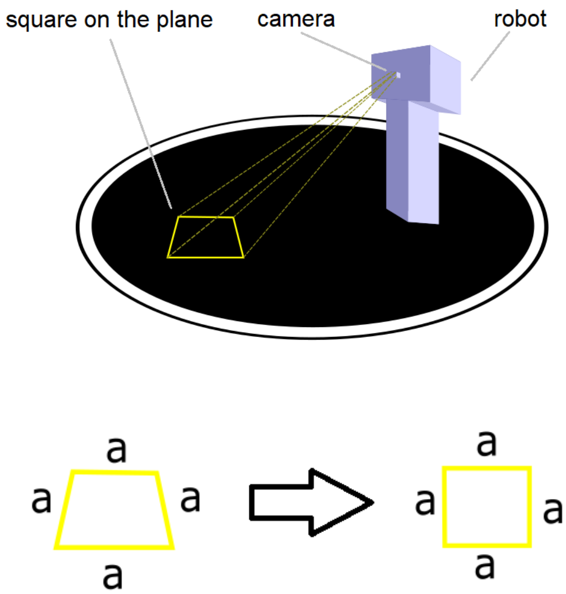


Figure 2. Transformation of the camera view into a plane parallel to the ring

tonomous, and all calculations are performed on a Raspberry Pi microcontroller or IntelNUC computer, which is located on its back. Melson took part in the international robot contests in Europe. It got silver medal in the humanoid sumo competition in Robot Challenge in Bucharest. The humanoid won also a lot of competition in Poland. However, attempts are made to improve this project, which will allow him to get more prizes.

3. Vision System

The usage of the vision system in robotics is becoming more and more popular. Thanks to proper analysis of the image provided by the camera, the robot is able to recognize its surroundings. This solution can be used in humanoid robot battles. For a robot on the battlefield, knowledge of its surroundings is important. It is practically impossible to use ordinary sensors, e.g. distances, because they can only return information about the location of other objects directly in front of the sensor. However, to know the location of the robot in the ring, it is important to be able to recognize the end lines of this field. Dohyō ring parameters are given before the fight, therefore, knowing its dimensions and colors, the vision system can be prepared before. A good way is to use a vision system that recognizes its position based on the image.

3.1 Hough Transform

There is a need for the robot to find a circle on the border of the battlefield - dohyō. This can help it to not get out of the ring and possibly find an opponent. A good solution for finding straight lines or circles in an image is to use the Hough Transform method. However, it requires a view of the ring from a direction perpendicular to it. It is therefore necessary to initially transform the image received from the camera, because the lines seen from the perspective of the robot's head are read as ellipses. However, having data

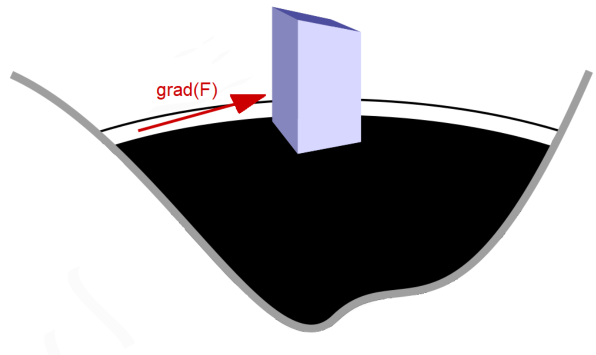


Figure 3. A color gradient tangent to the line surrounding the ring, where $grad(F)$ is gradient of F function representing pixel intensity

as to the position of the camera relative to the ground, the computer can project this image to the plane of the ring.

The camera is located on the head of the robot at a height of 25cm and is at an angle of 30 degrees to the plane parallel to the ring. Suppose there is a virtual square with a side length of 'a' on the plane of the ring. Then from the camera perspective it will be seen as a trapezoid. Inverse transformation can be made to project the entire image received from the camera onto the dohyō plane.

Once there is the ring line represented as a circle, it can be found using the Hough Transform method. It should be noted that the radius of the ring R is given, because in various competitions the radius of the ring is 80 cm. The first stage in Hough Transform is the initial separation of the outline of the objects in the image. The Laplacian function can be used on pixel intensity values. This will allow to extract the outlines of the objects in the image, because the lines around dohyō are white and the ring inside is black.

The equation describing a circle:

$$(x-x_0)^2+(y-y_0)^2=R^2, \tag{1}$$

where (x_0,y_0) are coordinates of the circle center, (x,y) are coordinates of points on the circle and R is given radius of the circle. Coordinates of a circle (x0, y0) are sought.

For each separated pixel, the so-called processed image creates the so-called Hough space. This is done by drawing circles with a radius R and the center of each extracted pixel, and the point where the most circles intersect in this space is most likely the center of the circle in the original image.

3.2 Location recognizing

By projecting the camera view onto the ring plane, it is able to find out the location of the robot. The image created after the transformation will be in the shape of a trapezoid, which can be interpreted as a view perpendicular to the axis of the cone of camera visibility. The tip of this cone is in the camera.

After casting the view on the ring and recognizing the opponent, the computer can create a map representing the robot's surroundings.

3.3 Identifying the opponent

The line recognition system can also be used to recognize the position of the opponent in the ring. Assuming that the robot will see the lines in the camera image, the opponent will most likely be

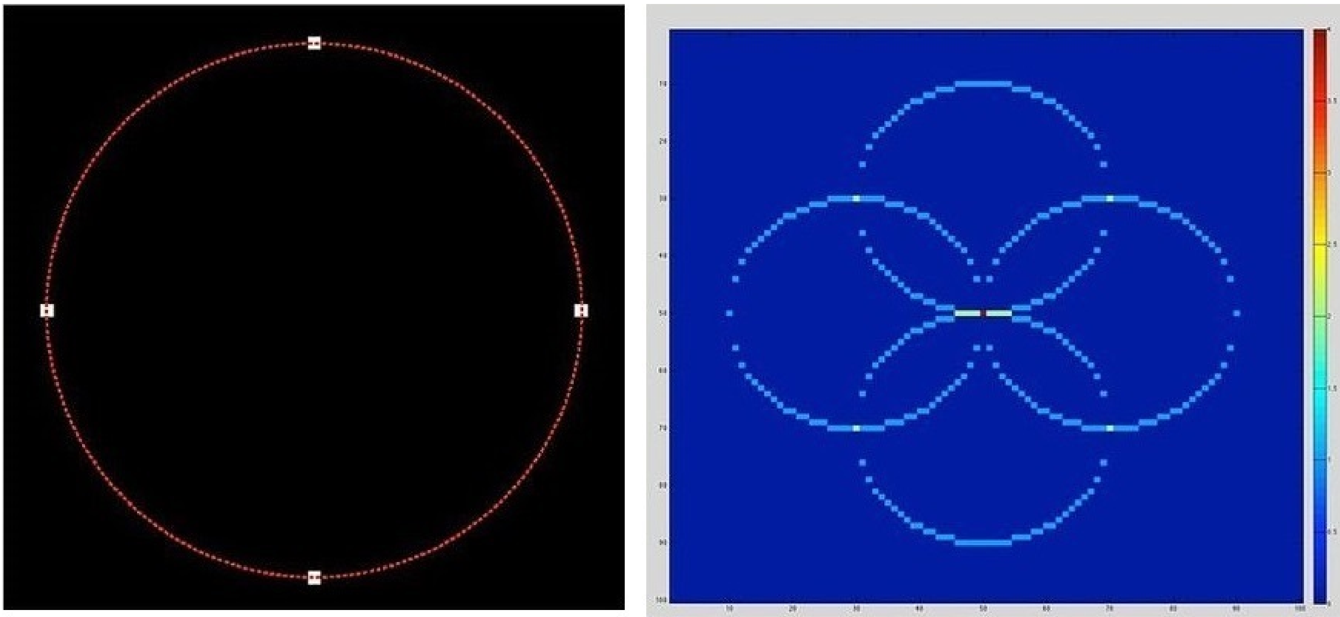


Figure 4. Finding a circle by the Hough transform method

where the line is broken. A good approach is to apply a tangent gradient to the line surrounding the circle. The function of the gradient of the tangent line to the circle is sought whose largest change is at the point where the ring line breaks.

After finding the white circle marking the end line of the ring, it is assumed that the opponent will break this line in the image. Therefore, the largest gradient is tangent to the circle line, and in the place where the gradient is the largest it is assumed that the opponent is located.

Assuming that the pixel intensity function is marked with white color F, its gradient is marked with the symbol grad (F).

Finding the opponent by the method of breaking the ring line will not get its exact position in the ring, but only the surroundings of its position. The place where the opponent may be located is illustrated in figure 3. Therefore, after finding the place where the line surrounding the ring is broken, a distance sensor attached to the servo is used, which rotates towards the arc, where the opponent is located. The servomechanism is attached to the abdomen of the robot. Only this sensor will provide accurate information about the opponent's distance to Melson.

4. Decision Making

The robot is able to execute some of the previously generated and saved trajectories. These trajectories allow it to do simple moves, for example: step forward or backward, rotation(15,30,45,60 or 90 degrees) or punching an enemy. The goal of the decision system is to choose moves, which helps the robot approach the opponent at minimum time. At the beginning of the battle the robot needs to find its location in the area. Basic assumption is that the global coordinate system is fixed at the center of the battlefield. After taking the first photo, the program places the Global Coordinate System (GCS) inside the circle. The first thing to do is to use the Hough Transform method to calculate a position of the protagonist and get the initial position of the robot.

After every step done by Melson, the program updates the position and orientation. It knows the previous position of the robot and distance passed, so computer can easy calculate the new one using vector of displacement. After that robot can find a position of the opponent in the GCS. During the first iteration robot calculate an angle of rotation needed for standing opposite to the enemy. In the second step the robot find opponent's new position. It allows Melson to calculate the vector of its antagonist's displacement. Next, robot can also predict the line which it will probably follow.

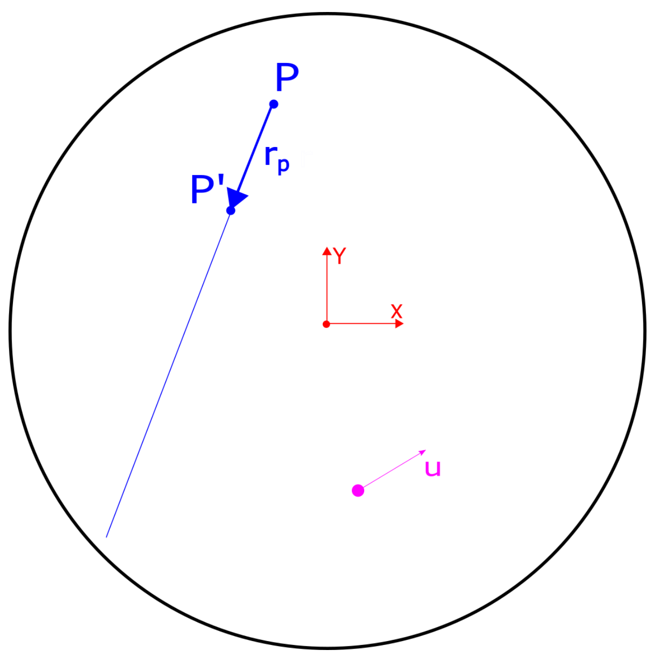


Figure 5. Finding the opponent. Vector r shows its displacement

When knowing the equation of that line it is easy to get a distance between Melson and that line (vector q). Also known is distance to the enemy (vector p). Using scalar product it is possible to calculate an angle between these vectors (β).

$$\beta = arccos(\frac{p \circ q}{|p||q|}) \quad (2)$$

The next step is to calculate an angle of rotation ϕ. After rotation unit vector u' should be collinear with the bisector of the angle ß. Then the robot moves one step forward.

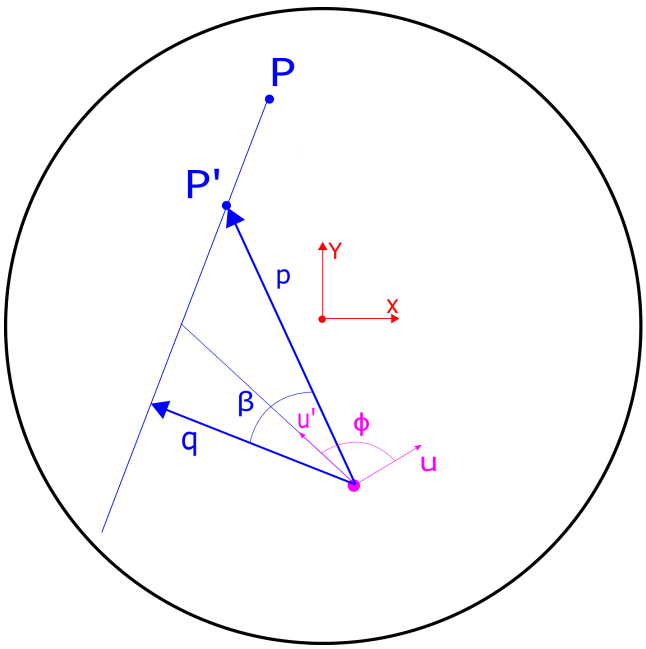


Figure 6. Angle of rotation ϕ

Melson repeats these operations. During every iteration it updates its position and opponent's walking line. When the robot stands less than 30 cm from the goal, it goes straight forward to approach the enemy and punch it. That method has two cases. First: the opponent is placed opposite to the robot. Knowing the enemy's behavior, the algorithm tries to reduce the distance between the two robots. Second: the opponent is placed backward and running away. Then the robot pursues it. Finally, Melson can punch the opponent in its back, which can be more effective.

5. Experiments

The experiment was decided to be divided into 3 stages:

1. Finding the end line of the ring and determining Melson's position on it.

2. Finding an opponent and determining its trajectory.

3. Making a decision about Melson's move relative to its opponent.

5.1 Experiment 1

At this stage of the experiment, the ring detection algorithm was examined based on individual photos taken with a camera placed on Melson's head. Tested on the basis of two photos shown of fig. 8:

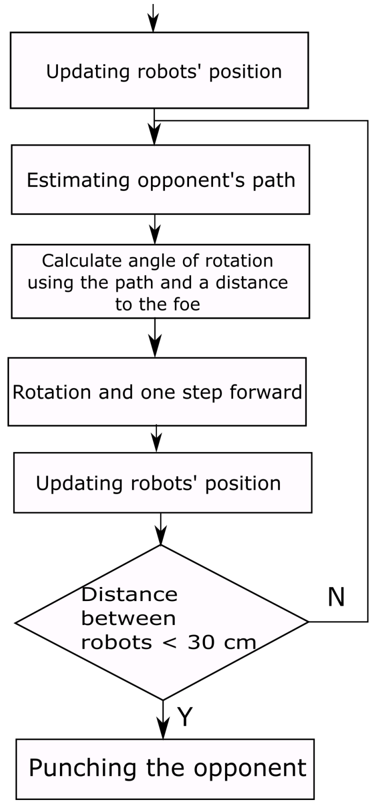


Figure 7. Block diagram of the decision system

As a result of the transformation of the photo into a view perpendicular to the ring and using the Hough transformation to find the best-fit circle, the photos presented in figure 8 were obtained shown together with raw photos. The found circle is marked in green, the center of the circle as a red point, and the blue lines are the camera's visibility. The circles indicate more or less the good location of the real circle. The relative approximation error is approximately 3%.

The above transformation of the photo into a view parallel to the ring indicates the border of the image cone that the camera sees. The camera lens is located at the intersection of two lines. On this basis, Melson can find its location:

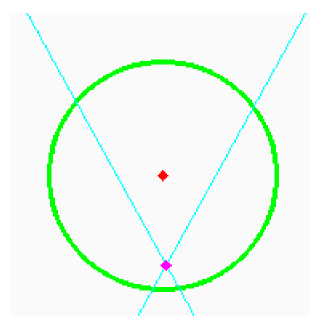


Figure 9. Calculation of the robot's position based on the photo of the ring

Melson's position is marked in pink. The true location of the humanoid also differs from the predicted one by about 3%. The results are satisfactory.

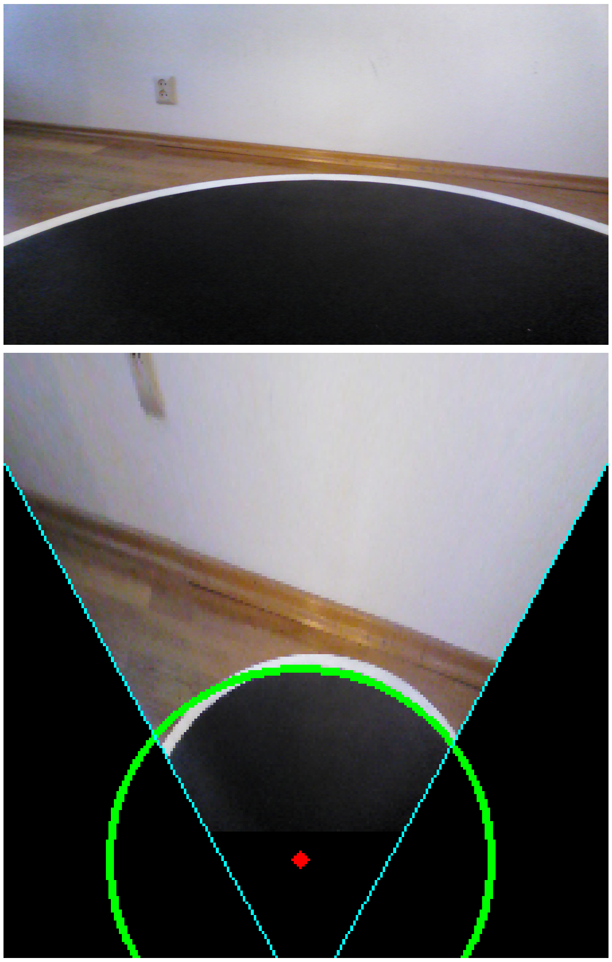


Figure 8. Transformation of the photo into a view parallel to the ring

5.2 Experiment 2

During the experiment, a small plastic model of the KUKA robot was used as the opponent which Melson is looking for in the ring. For this purpose, the tangent gradient method to the line surrounding the ring is used, described in the previous chapter.

At this stage of the experiment, it is assumed that Melson is stationary and his opponent moves. Every now and then (approximately the time for Melson to take one step) a picture of the surroundings is taken and the opponent is sought. In the case of this experiment, it was decided to examine its motion based on three photos taken every 3 seconds.

As shown in the figure 10, the robot moves to the right and each time crossed the white arc surrounding the ring. These images were obtained after the Hough transformation.

In figure 10 the robot was found in a good surroundings of its true location. The circle found was marked in green, the center of the circle was marked as a red point, the light blue lines were the camera visibility, and the blue point was the expected location of the opponent. The relative approximation error is 3%.

5.3 Experiment 3

To plan a path the robot needs two different positions of the opponent. Using that information the program extrapolates the line

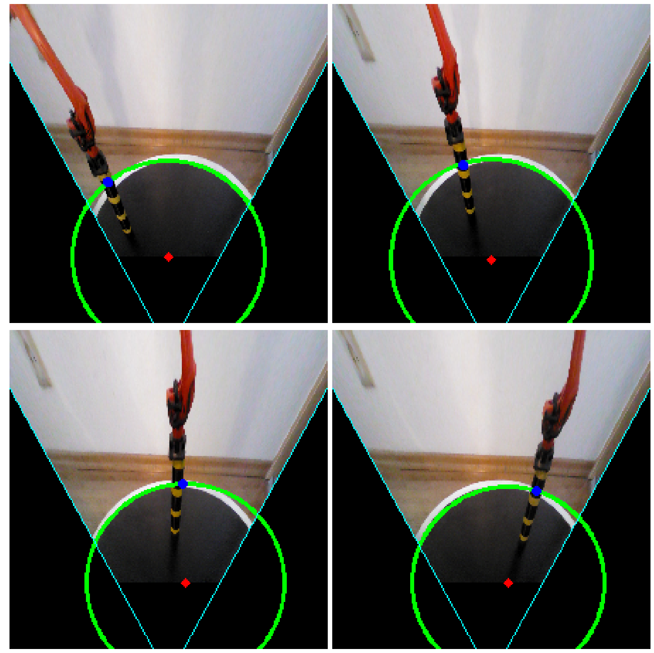


Figure 10. Calculation of the opponent's moving position

followed by the KUKA model. After estimating the equation of the line the robot calculates the distance to it. The error of that operation is approximately 6%. The program also calculates a vector between the robot and the opponent using vectors of the robots' positions. The calculated length of that vector differs from the real one by 3%.

The last step is to calculate an angle the robot needs to turn as it was shown in the "Decision making" paragraph. The Melson cannot turn a custom angle. It has to choose one from this list: 15,30,45,60 or 90 degrees. The algorithm executes the one, which is the nearest to the desired angle ϕ. In this example the angle was $12°$ clockwise, so the robot rotated through $15°$.

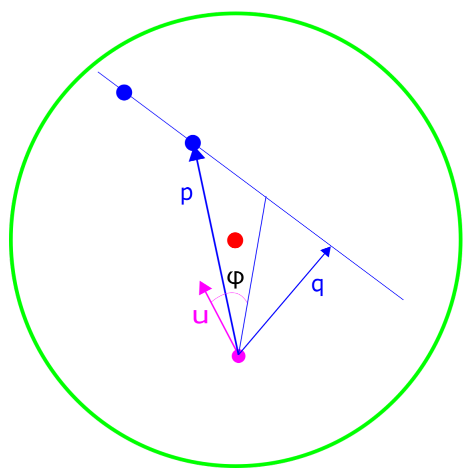


Figure 11. Result of the experiment. The robot should rotate clockwise through the angle ϕ

6. Conclusions

Humanoids are becoming an increasingly popular topic in robotics. More and more scientists, students and engineers deal with them, so it is important to develop an appropriate system of their operation. Using the camera and appropriate image analysis the robot can get a lot of information about its environment. A good example is the Hough transformation, which can help recognize shapes, e.g. circles, as shown in this article. After obtaining information about the space around the humanoid, the robot can go to the decision-making system. It will help the robot behave adequately to the dynamically changing area, as illustrated by the example of sumo competition.

Acknowledgments

Authors are members of Students' Robotics Association (Koło Naukowe Robotyków) at the Faculty of Power and Aeronautical Engineering, Warsaw University of Technology. Research was done co-operating with the Faculty of Power and Aeronautical Engineering within the Framework of "Najlepsi z Najlepszych 4.0!" program.

References

[1] C. R. Torrico, A. B. Leal, and A. T. Watanabe, "Modeling and supervisory control of mobile robots: A case of a sumo robot," *IFAC-PapersOnLine*, vol. 49, no. 32, pp. 240 – 245, 2016. Cyber-Physical Human-Systems CPHS 2016.

[2] H. Erdem, "Application of neuro-fuzzy controller for sumo robot control," *Expert Systems with Applications*, vol. 38, no. 8, pp. 9752 – 9760, 2011.

[3] J. Zhang and S. Singh, "Loam: Lidar odometry and mapping in real-time," July 2014.

[4] K. Okada, M. Inaba, and H. Inoue, "Walking navigation system of humanoid robot using stereo vision based floor recognition and path planning with multi-layered body image," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 3, pp. 2155–2160 vol.3, 2003.

[5] C. O. Vilão, D. H. Perico, I. J. Silva, T. P. D. Homem, F. Tonidandel, and R. A. C. Bianchi, "A single camera vision system for a humanoid robot," in *2014 Joint Conference on Robotics: SBR-LARS Robotics Symposium and Robocontrol*, pp. 181–186, 2014.

Validation of Distributed State Estimation for Localization of Small Satellites and Swarm Formation

Alexander Frye[1], Hadi Fekrmandi [2], Randy C. Hoover[1], and Amir Hossein Tamjidi[3]

[1]Department of Computer Science and Engineering, South Dakota School of Mines & Technology, 501 E. Saint Joseph St., Rapid City, SD 57701, Email: Alexander.Frye@mines.sdsmt.edu, Randy.Hoover@sdsmt.edu

[2]Department of Mechanical Engineering, South Dakota School of Mines & Technology, 501 E. Saint Joseph St., Rapid City, SD 57701, Email: Hadi.Fekrmandi@sdsmt.edu

[3]Software Engineer, Driving Algorithms Team, Chess Dr, Foster City, California 94404, Email: ahtamjidi@gmail.com

ABSTRACT

The objective of the Swarm Demonstration for Space Autonomous Tasks (SD-SAT) project is to demonstrate proximity formation flying of a swarm of CubeSats using vision-based relative camera sensors. In this paper, a linear model of orbital dynamics and a state-feedback controller were implemented for proximity formation flying in low-earth orbit satellite simulations. The model was tested and proven to accurately represent the nonlinear orbital dynamics equations. The controller was able to maneuver a group of three cube-satellites into the desired formation. The method is shown to be adaptable and scalable for N number of satellites in proximity operation missions. Development and integration of the proposed method into the vision-based guidance, control, and navigation subsystems will pave the way for new attitude determination and control technology for swarm proximity formation flying missions.

1. Introduction

According to NASA's Small Spacecraft Technology Program (SSTP), a spacecraft is called a "small spacecraft" when its mass is below 180 kg. CubeSats are a class of small spacecrafts called nanosatellites weighting in the range of 1-10 kg. CubeSats are built to standard dimensions known as Units (i.e. 1U, 2U, 3U, or 6U) where each unit is a 10 cm x 10 cm x 10 cm cube [1]. CubeSats are suitable for formation flying missions because of their modular nature and low cost [2]. Multiple satellites working together can offer additional capabilities over monolithic satellites due to their cooperative behavior within the swarm. Theses advantages include increased flexibility, accuracy, maximum range, significantly enhanced flexibility (adaptability, scalability, evolvability, and maintainability), and robustness (reliability, survivability, and fault tolerance) [3]. The capabilities needed to support swarm missions go beyond operator-specified geometry, alignment, or separation, but also include cross-link communication while maintaining position in the formation [4]. Table 1 shows a few of the ongoing and recently completed scientific missions by a swarm of CubeSats.

Table 1. Scientific research missions by swarm of CubeSats missions

Mission Name	Mission Objective
CPOD - CubeSat Proximity Operations Demonstration [5]	Poximity operations and docking of two 3U CubeSats
Frisbee – A Platform for Small Satellite Science Swarms [6]	Space Weather Advanced Research Mission (SWARM)
Olfar - Swarm Telescope Constellation [7]	A Radio Telescope Based on Nano-Satellites in Moon Orbit
EDSN - Edison Demonstration of SmallSat Networks [8]	Intersatellite Communications Architecture
Nodes - Networked spacecraft command and control [9]	synchronized, multipoint scientific measurements
HelioSwarm - Swarm Mission Design in High Altitude Orbit [10]	Studying Physics of Space Plasma Turbulence
CLICK - CubeSat Laser Infrared CrosslinK Mission [11]	Optical Communications Cross-link communication

In many scientific missions, particularly when accurate spatial and temporal data must be provided, accurate state estimation is required to satisfy the mission objectives. For example, simultaneous measurements from a 3-dimensional volume of space could be made feasible by a swarm and these measurements are advantageous for a variety of studies in space physics and Earth science. Swarm Orbital Dynamics Advisor (SODA), is a platform developed by NASA that takes general configuration commands and provides the orbital maneuvers needed to achieve the desired formation flying through swarm relative motion (shown in Figure 1).

Although there has been promising progress in recent years in the field of swarms of CubeSats, the current state of the art of guidance navigation and control (GNC) relies particularly on the Global Positioning System (GPS). The performance of the GPS subsystem was evaluated in a mission to study space weather measurements during the Radio Aurora explorer II (RAX-2) CubeSat mission [13]. The results presented a the GPS-derived error of 4.02 m for position and 0.48 m/s for velocity.

While the GPS subsystem may have sufficient accuracy for providing useful positioning information for some small satellite swarm missions (for example weather data measurements from distributed points of the atmosphere), many scientific missions require much higher position accuracy. Examples include those involving multi-satellite docking, earth and space imaging, and inspection missions around larger satellites. One of the fundamental drawbacks of

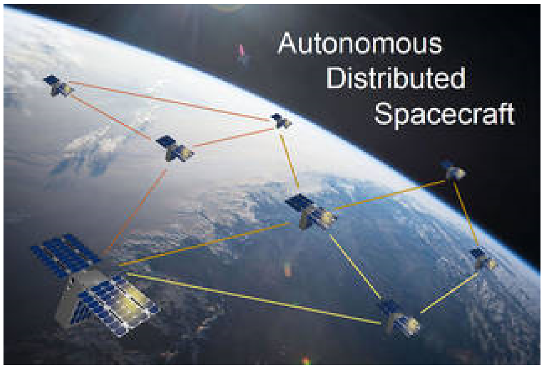


Figure 1. Swarm Orbital Dynamics Advisor (SODA) controls the configuration of satellites in a distributed swarm mission [12] (Figure courtesy of NASA Ames Research Center).

current GNC techniques is their reliance on a centralized process using global positioning system (GPS) [14]. While real-time positioning computed by standard GPS service is adequate for some dispersed applications (i.e. constellation missions), inherent position discontinuities are not acceptable for proximate formation flying missions demanding high-precision science instruments for simultaneous measurements [15]. In addition, deep space swarm missions in GPS-denied environments will rely on alternative orbit determination using radio or optical-based measurements [12]. Optical techniques such as Smart Phone Video Guidance (SVGS) [16] have been considered for obtaining an accurate relative state estimation of the members of the swarm and offer an efficient relative state estimation in close proximity. Prior work using visual-based systems for accurate state/pose estimation has shown promising results to ground-based robotic systems (UGVs) operating in GPS denied environments [17]. Use of an illuminated fiducial system (referred to as an AR or Augmented Reality Tag was studied for proximity formation flying [18]). In this method, a unique identifier is allocated to each CubeSat within the swarm and it is shown that the vision-based pose estimator provides relative spacial pose estimation that is highly accurate compared to GPS In the current paper, a computationally efficient controller is proposed for CubeSat proximity formation flying missions. It is assumed that the relative states are available through both GPS and vision-based methods. The challenge is for agents for maintain certain relative formation and recover the disturbances imposed due to the presence of low density atmosphere in low earth orbit (LEO). The proposed system provides a solution for maneuvering CubeSats into the desired formation in LEO. The proposed method of control may also be developed further to completely rely on relative measurements for deep space applications where accurate positioning of CubeSats is important for multiple-node sensor measurements and mapping.

2. GUIDANCE, NAVIGATION and CONTROL

To control cube-satellites in orbit around earth, a linearized model and controller are required. To test the linear model and controller, an accurate, nonlinear model for orbital dynamics was implemented. This nonlinear model is a frictionless, single-body model of gravitational acceleration. These limitations shouldn't effect the simulation, as in a low number of orbits the loss in velocity from air friction is negligible, and as long as the satellites are not near the moon there are no other bodies that would significantly alter their trajectories.

The acceleration due to gravity from any body to another is

$$a_g = \frac{Gm_1}{r^2}\frac{\vec{r}}{r}, \text{ or } a_g = \frac{\mu\vec{r}}{r^3},$$

where G is the gravitational constant ($6.673x10^{-11}$ Nm^2/kg^2), m_1 is the mass of the orbited body (for earth, $5.972x10^{24}$ kg), μ is the standard gravitational parameter (for earth, $3.986x10^{14}$ m^3/s^2), r is the distance in meters between the center of mass of each body, and $\vec{r}$ is the vector from the satellite to the center of mass of the orbited body. This model is accurate, but not entirely useful because as a spacecraft orbits, the radius vector $\vec{r}$ is constantly changing. To solve this, the equation can be re-written Cartesian coordinate system in terms of the $[x, y, z]$, where the radius can be redefined as

$$\vec{r} = \vec{x} + \vec{y} + \vec{z} \text{ and } r = \sqrt{x^2 + y^2 + z^2}.$$

Now that there is a nonlinear reference model, candidate linear models can be tested. The linear model used (Equation 1) is based on the model created by [19], with small changes to the matrix D_0 because that model did not work as intended. 0_3 is a 3 by 3 matrix of zeros, I_3 is the dimension 3 identity matrix, and μ and r are body and orbit-specific constants defined above. Changing μ affects what body the model pertains to, and changing r changes the radius of orbit it is linearized about.

$$\begin{gathered} A_0 = \begin{bmatrix} 0_3 & I_3 \\ D_0\mu/r^3 & S_0\sqrt{\mu/r^3} \end{bmatrix} \qquad B_0 = \begin{bmatrix} 0_3 \\ I_3 \end{bmatrix} \\ D_0 = \begin{bmatrix} -3 & 0 & 0 \\ 0 & -3 & 0 \\ 0 & 0 & -1 \end{bmatrix} \qquad S_0 = \begin{bmatrix} 0 & 2 & 0 \\ -2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \end{gathered} \tag{1}$$

This model is linearized about an equatorial, circular orbit, but as long as the orbital plane remains constant. If a polar orbit (an orbit not in the plane of the equator) was required, the axes could be rotated to bring that orbit into the $[x, y]$ The state vector for this model is given by

$$\mathbf{x} = [x \;\; y \;\; z \;\; \dot{x} \;\; \dot{y} \;\; \dot{z}]^T,$$

which comprises the position and velocity data in $[x, y, z]$ but contains no rotation or angular velocity data. This model is for a single satellite, and said satellite's velocity and acceleration can be modeled with $\dot{\mathbf{x}} = A_0\mathbf{x} + B_0\mathbf{u}$. To validate the usage of a linearized version of the nonlinear dynamics for the purposes of swarm control, the researchers simulated each model with identical initial conditions and plotted the result. This test involves plotting several satellites in orbits around earth, with realistic orbit radius and velocity. When an uncontrolled satellite ($\mathbf{u} = 0$) is plotted in a circular orbit of radius 6,317 km (100 km above the earth's surface) with an orbital velocity of 7,848 m/s (the velocity of a circular orbit at this height), the linear model perfectly recreates the nonlinear dynamics, even with small initial positional offsets from the ideal orbit (Figure 2).

To create a model for multiple satellites in a swarm, the A_0 and B_0 matrices from Equation 1 can be combined, with the assumption that the dynamics of each satellite are completely independent from one another (this holds so long as the satellites do not thrust directly at one another while in close proximity).

The state matrices A and B are then

$$A = I_n \otimes A_0,$$

$$B = I_n \otimes B_0,$$

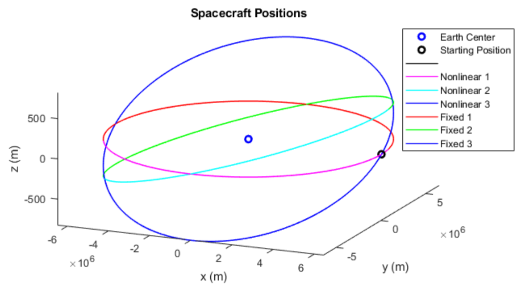


Figure 2. Performance of the linearized vs. nonlinear models in three difference orbit

and **x** is then a vector of n x_0 blocks, for example if $n = 3$

$$\mathbf{x} = [x_0 \ x_0 \ x_0]^T .$$

In these equations, $\otimes$ is the Kronecker product and I_n is the identity matrix of dimension n. For a 3-satellite system: **x**, A, and B can be created using the previous equations where $n = 3$. In block matrix form, with 0_n being an $n \times n$ matrix of zeros and $0_{n,m}$ being an $n \times m$ matrix of zeros, **x**, A, and B are

$$\mathbf{x} = \begin{bmatrix} x_0 \\ x_0 \\ x_0 \end{bmatrix} \quad A = \begin{bmatrix} A_0 & 0_6 & 0_6 \\ 0_6 & A_0 & 0_6 \\ 0_6 & 0_6 & A_0 \end{bmatrix} \quad B = \begin{bmatrix} B_0 & 0_{6,3} & 0_{6,3} \\ 0_{6,3} & B_0 & 0_{6,3} \\ 0_{6,3} & 0_{6,3} & B_0 \end{bmatrix}. \tag{2}$$

From this model, a state feedback controller was designed to stabilize the relative positions of these satellites in close proximity to each other both in terms of position and velocity. This controller is based off of a PD controller, with the main goal of being fuel efficient and having a low thrust magnitude. Adding an integral term could potentially reduce the steady-state position error, but tends to add oscillations and could potentially destabilize the system. The full state equation is now: $\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} = A\mathbf{x} + BK(\mathbf{x} - g)$ where g comprises the desired positions of satellites 2 and 3 relative to satellite 1 and K is the gain matrix. The state feedback equation K is described in Equation 3, where $0_{n,m}$ is an $n \times m$ matrix of zeros and $k_p = -0.000001$ and $k_d = -0.001$. These gains are extremely small because the maximum thrust of a cube-satellite is very limited, especially when using high-efficiency propulsion mechanisms such as ion thrusters.

$$K_0 = \begin{bmatrix} k_p I_3 & k_d I_3 \end{bmatrix} \quad K = \begin{bmatrix} 0_{9,6} & 0_{9,6} & 0_{9,6} \\ -K_0 & K_0 & 0_{3,6} \\ 0_{3,6} & 0_{3,6} & 0_{3,6} \\ -K_0 & 0_{3,6} & K_0 \end{bmatrix}. \tag{3}$$

3. RESULTS

To demonstrate this controller, numerous tests were conducted with a 3-satellite formation. Every test had one satellite (hereafter referenced as satellite 1) that started orbiting in a circular 100 km above earth, with the other two satellites (satellites 2 and 3) starting in near-identical orbits, just 100 meters away from satellite 1 in the X, Y, and Z directions.

In one specific example, satellites 2 and 3 had initial and desired positions detailed in Table 2. This example was selected because it shows that the controller is capable of moving the satellites with very little thrust from a loose to a tight formation. The satellites are controlled to hold their positions relative to the global coordinate system (holding their positions relative to a distant observation target such as a planet of star).

Table 2. Positions in meters relative to satellite 1

	X	Y	Z
Satellite 2 start	100	0	0
Satellite 3 start	-100	0	100
Satellite 2 desired	-10	-10	0
Satellite 3 desired	-10	10	0

First, the uncontrolled system was plotted. The satellite positions plotted around the earth (Figure 3) shows them orbiting, but because of the incredible orbit radius there is no indication of their separation. In contrast, when the positions of satellites 2 and 3 are plotted relative to satellite 1 (Figure 4) they form an odd pattern. This can be better understood when looking at each axis overlaid, and plotted vs. time, as they are in Figure 5. It should be noted that a 100 km orbit of earth has an orbit period of 5,300 seconds (just under one and a half hours). In these figures the satellites oscillate from their start position to the direct opposite position relative to satellite 1 and back every orbit. They don't stay in the same formation, but neither do they spread out away from each other.

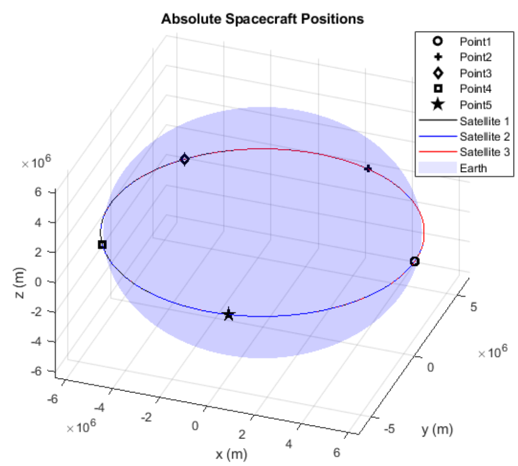


Figure 3. Absolute positions of satellites 1, 2, and 3 (uncontrolled).

In contrast to the uncontrolled test, the controlled plots have no oscillatory behavior. The plot of the absolute positions (Figure 6) of each satellite is very similar to the uncontrolled plot, again because of the extreme distances involved. In contrast, the positions of satellites 2 and 3 relative to satellite 1 are controlled to slowly travel to the goal positions (Figure 7). This same phenomena can also be seen in Figure 8.

4. CONCLUSION

The swarm of the small satellites are capable of delivering substantial science return, at a portion of the cost of larger counterparts. The proximity formation flying of swarm of CubeSats is particularly challenging because of the limitations in propulsion, power and computational capabilities of small spacecrafts due to their small size. In this work, these shortcomings are addressed by development of an accurate and robust guidance, navigation and control

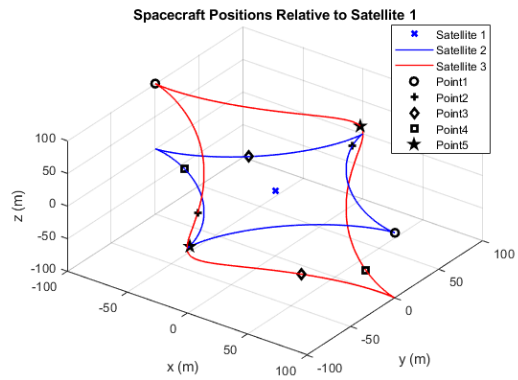


Figure 4. Positions of satellites 2 and 3 relative to satellite 1 (uncontrolled).

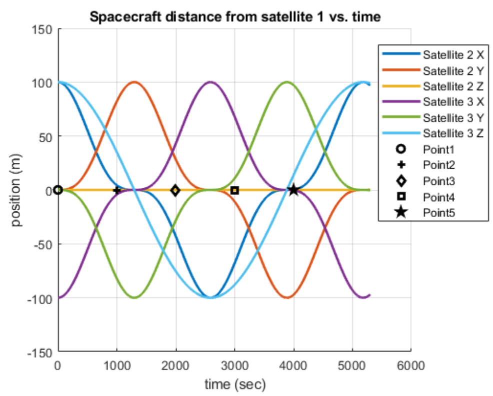


Figure 5. Positions of satellites 2 and 3 relative to satellite 1 (uncontrolled).

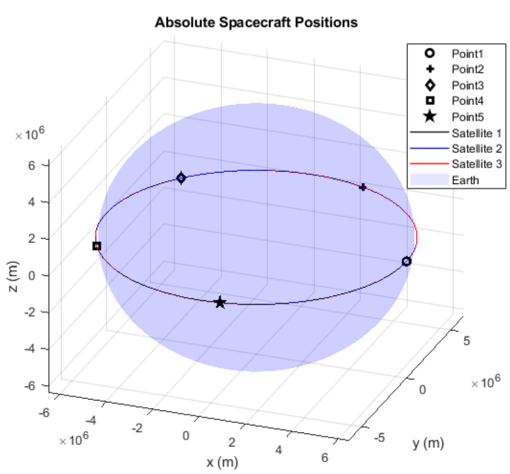


Figure 6. Absolute positions of satellites 1, 2, and 3 (controlled)

subsystem. The proposed method of this study using the linear model of orbital dynamics and a low cost state-feedback controller for swarm has shown promising performance. This will enable the agents of swarm work collaboratively to deliver the high quality scientific measurement. Moreover, for missions beyond the low earth orbit, two key technical challenges for realizing spacecraft swarms

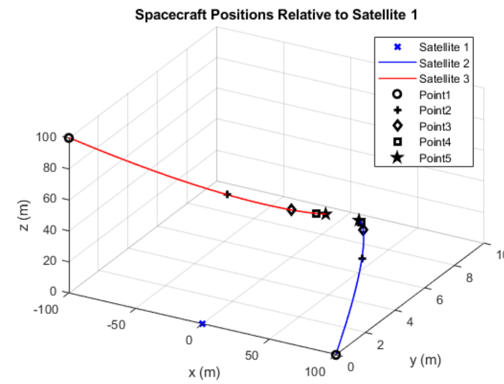


Figure 7. Positions of satellites 2 and 3 relative to satellite 1 (controlled)

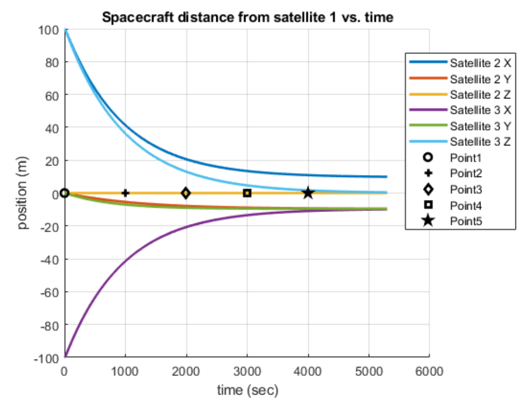


Figure 8. Positions of satellites 2 and 3 relative to satellite 1 (controlled)

in GPS-denied environments of deep space that has been tackled include: (1) fast and stable inter-connectivity among the spacecraft using only relative state measurement and (2) adequate state estimation to accurately and continuously determine the exact position of the ships forming the swarm.

5. ACKNOWLEDGMENT

Authors would like to express gratitude to Dr. Edward Duke, director of South Dakota NASA EPSCoR program for funding this research under Research Initiation Grant (RIG). Also, we greatly acknowledge both John Rakoczy and Brian Bae from control systems design and analysis division of NASA Marshall Space Flight Center for their countless feedback and support of this project.

References

[1] J. Crusan and C. Galica, "Nasa's cubesat launch initiative: Enabling broad access to space," *Acta Astronautica*, vol. 157, pp. 51–60, 2019.

[2] F. Y. Hadaegh, S.-J. Chung, and H. M. Manohara, "On development of 100-gram-class spacecraft for swarm applications," *IEEE Systems Journal*, vol. 10, no. 2, pp. 673–684, 2016.

[3] S. Bandyopadhyay, G. P. Subramanian, R. Foust, D. Morgan, S.-J. Chung, and F. Hadaegh, "A review of impending small satellite formation flying missions," in *53rd AIAA Aerospace Sciences Meeting*, p. 1623, 2015.

[4] J. Cockrell, R. Alena, D. Mayer, H. Sanchez, T. Luzod, B. Yost, and D. Klumpar, "Edsn: A large swarm of advanced yet very affordable, cots-based nanosats that enable multipoint physics and open source apps," 2012.

[5] C. W. Roscoe, J. J. Westphal, and E. Mosleh, "Overview and gnc design of the cubesat proximity operations demonstration (cpod) mission," *Acta Astronautica*, vol. 153, pp. 410–421, 2018.

[6] A. da Silva Curiel, M. Meerman, D. Liddle, S. Schwartz, and M. Sweeting, "Frisbee–a platform for a small satellite science swarms," 2003.

[7] S. Engelen, C. J. Verhoeven, and M. J. Bentum, "Olfar, a radio telescope based on nano-satellites in moon orbit," 2010.

[8] J. Hanson, J. Chartres, H. Sanchez, and K. Oyadomari, "The edsn intersatellite communications architecture," 2014.

[9] J. Hanson, A. G. Luna, R. DeRosee, K. Oyadomari, J. Wolfe, W. Attai, and C. Prical, "Nodes: A flight demonstration of networked spacecraft command and control," 2016.

[10] L. Plice, A. Dono Perez, and S. West, "Helioswarm: Swarm mission design in high altitude orbit for heliophysics," 2019.

[11] K. Cahoy, P. Grenfell, A. Crews, M. Long, P. Serra, A. Nguyen, R. Fitzgerald, C. Haughwout, R. Diez, A. Aguilar, *et al.*, "The cubesat laser infrared crosslink mission (click)," in *International Conference on Space Optics—ICSO 2018*, vol. 11180, p. 111800Y, International Society for Optics and Photonics, 2019.

[12] T. Conn, A. Perez, L. Plice, and M. Ho, "Operating small sat swarms as a single entity: Introducing soda," 2017.

[13] J. Arlas and S. Spangelo, "Gps results for the radio aurora explorer ii cubesat mission," in *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, p. 123, 2013.

[14] L. Winternitz, B. Bamford, S. Price, A. Long, M. Farahmand, and R. Carpenter, "Gps navigation above 76,000 km for the mms mission," tech. rep., NASA Goddard Space Flight Center, Greenbelt, MD, United States, 2016.

[15] A. Richardson, J. Strom, and E. Olson, "AprilCal: Assisted and repeatable camera calibration," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, November 2013.

[16] C. Becker, R. Howard, and J. Rakoczy, "Smartphone video guidance sensor for small satellites," in *Proceedings of the AIAA/USU Conference on Small Satellites*, pp. 1–8, Aug 2013.

[17] R. C. Hoover, *POSE ESTIMATION OF SPHERICALLY CORRELATED IMAGES USING EIGENSPACE DECOMPOSITION IN CONJUNCTION WITH SPECTRAL THEORY*. PhD thesis, Colorado State University, 2009.

[18] H. Fekrmandi, S. Rutan-bedard, A. Frye, M. Yoon, and R. Hoover, "Vision-based guidance and navigation for swarm of small satellites in a formation flying mission," 2019.

[19] F. Hadaegh, B. Acikmese, D. Bayard, G. Singh, M. Mandic, S. Chung, D. Morgan, K. Alfriend, M. Akella, J. Hurtado, *et al.*, "Guidance and control of formation flying spacecraft: From two to thousands," *Adventures on the Interface of Mechanics and Control*, pp. 327–371, 2012.

Revisiting the Pull Test for the Measurement of Static Coefficient of Friction

Oren Masory
Department of Ocean and Mechanical Engineering
Florida Atlantic University, Boca Raton, FL, USA
masoryo@fau.edu

Abstract
Measuring static coefficient of friction (SCOF) by pull test, in particular in wet conditions, is considered by many unreliable. This paper describes the results of 2880 pull tests that were performed by 12 teams of inexperienced individuals. Each team tested 2 tiles under three conditions: Dry, Wet (tap water) and Soaped water. For each case 10 pulls in the 4 cardinal directions (total of 40) were taken. The results indicate that if the tests are executed correctly (according to ASTM C-1028) the SCOF in the wet conditions are well distinguished from the results obtained in dry conditions. Obviously, the results for the SCOF for the soaped condition were even lower than the one for the wet conditions.

I. INTRODUCTION

The first Pull Test, for the determination of SCOF, was performed by Leonardo da Vinci at the end of the 15th century [1, 2] and over the years it was formulized by Amontons, Coulomb and others [3]. Pull test were used for many years to determine the SCOF of walking surfaces and several standards, describing the procedure by which the test has to be executed, were established [e.g. 4-6].

This test method was criticized, in particular, on two issues [7]:

1. Repeatability – Since the test is performed manually, the concern is that the same operator introduces error due his inability to repeat the test. Or, in case of more than one operator the results will not be same due to the way the individual performs the test.
2. Test of wet surfaces – While the test's procedure is acceptable for dry surfaces, the concern is that when wet surface is tested the results are affected by a phenomena called *sticktion*, which results in higher values of SCOF for wet surface than of dry surface.

Sticktion, as applied to wet testing of walking surfaces, arises as a function of the test slider's contact time on the surface. If there is any delay between the instant of slip meter pad contacts the surface and the application of the horizontal force (the pulling force) *sticktion* will occur. When a shoe hits a walking surface the interaction time is very short. On wet surfaces, when the contact time shorter than 0.2 seconds (200 milliseconds) significant *sticktion* is produce that can result in slip meter readings that are higher under wet conditions than would be obtained on the same surface in a dry state.

Some commercial slip meters such as English XL, BOT-3000, British Pendulum Tester (BPT), Brungraber Mark II and MARK IIIB and others claim to resolve the above issues. These devices requires minimum training and the value of the measured SCOF is immediately and explicitly provided to the user. Publication that compared the performance of these devices indicate large discrepancies between the SCOF values obtained by the different devices [8-20].

This paper revisits the commonly used pull test and provides the results of a large number of pull tests which might give some insight to the issues above.

II. TEST PROCEDURE

48 inexperienced individuals, with little background in engineering, were randomly divided to 12 team of 4 members. Each team tested 2 tiles (see Table 1). Each team was provided with a 3" square 3/8" thick Neolite pad (ASTM D2000 rating AA and 70A shore hardness), a 20[lb] weight and a force gauge (the gauge did not hold the peak reading), bottle of tap water and a bottle of tap water with 0.0625% of commercial detergent.

Six different tiles were used in these experiments (see Figure 1):

1. Ceramics
2. Vinyl
3. Porcelain with fine sandy texture
4. Glazed ceramics with grooves
5. Polished marble
6. Porcelain with smooth surface.

Table 1: Tiles assigned to each team.

Team	Tiles	Team	Tiles
1R	1,5	4R	2,4
1T	3,5	4T	2,6
2R	2,4	5R	5,6
2T	1,2	5T	1,5
3R	3,6	6R	3,4
3T	1,6	6T	3,4

The instructions provided to the teams were simple:

a. Calibrate the force gage using accurate weights.
b. Measure the weight, used as a load, using an accurate electronic scale.
c. Place the pad on the tile and mount the weight on top the pad.
d. Pull the assembly parallel to the tile's surface and measure the force needed to initiate a motion.
e. For the wet and "soaped" surfaces make sure to flood interface between the pad and the tile and perform the test as soon as possible after the flooding.
f. Repeat the test 10 times in four perpendicular direction of the tile (total 40 reading for each tile)
g. Perform (e) for the conditions: Dry (D), Wet with water (W) and with soaped water (S).

Thus, 160 pull tests were performed on each tile for each of the conditions Dry (D, Wet with water (W) and wet with soaped water (S). In total 2,880 pull tests were carried out in this experiment.

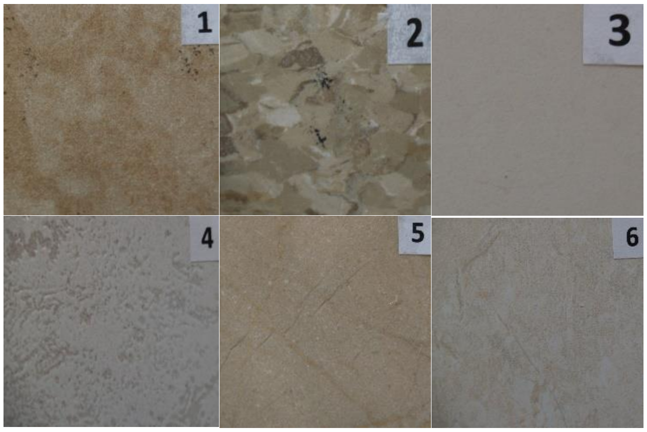


Figure 1: Photos of the 6 tiles.

III. COF TESTS' RESULTS

The results, using all data are shown in Figure 1. It shows the average value of the SCOF obtained for each tile (160 readings for each team).

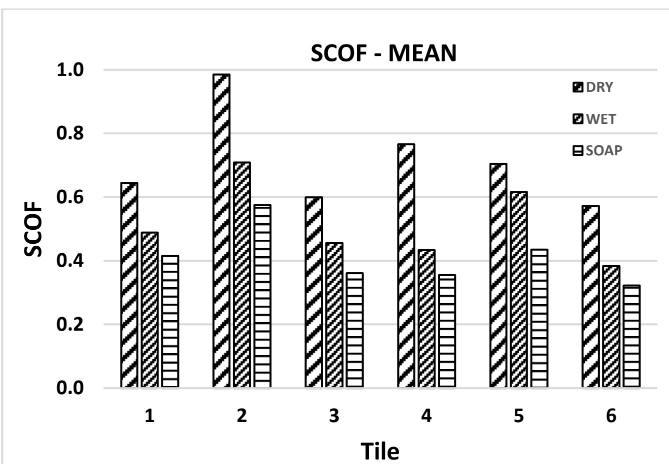


Figure 2: SCOF's mean for the six tiles

The results obtained for all tiles in the respected conditions are the expected ones. Meaning that the SCOF on water wetted surface is lower than in dry conditions and higher than the one with soap.

However, these results contradicts the commonly accepted concern that due to *stickion* the reading of the SCOF in wet condition using pull test will be higher than the reading in dry conditions.

The standard error for each of the cases was calculated and is shown in Table 2. As shown the standard error is very small compared to the mean value of the SCOF. Also, for all tests, the largest values of the standard error are in the pull tests under dry conditions. This due to the fact that the peak value of the pulling force is not recorded automatically by the force gauge and has to be "captured" manually.

Table 2: Standard error for the SCOF of all tiles and test conditions,

Tile	DRY	WET	SOAP	ALL
1	0.0148	0.0098	0.0102	0.0116
2	0.0211	0.0114	0.0113	0.0146
3	0.0125	0.0069	0.0069	0.0087
4	0.0169	0.0105	0.0088	0.0120
5	0.0121	0.0099	0.0072	0.0097
6	0.0114	0.0113	0.0061	0.00961

Figures 3 show results obtained for each tile and by each team and Table 3 provides the corresponding standard error for each case.

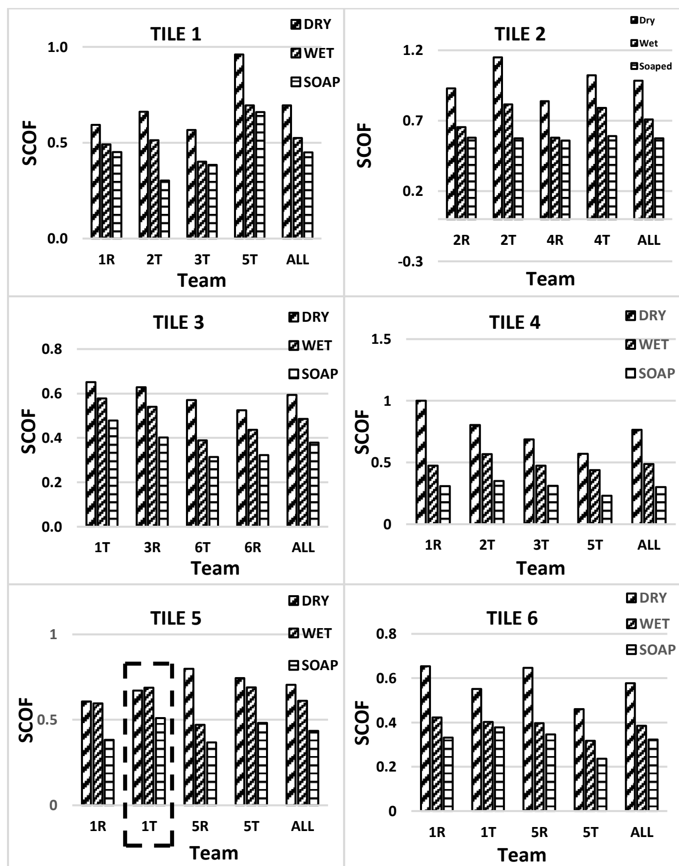


Figure 3: Tests' results for all tiles.

From Figure 3 and the values in Table 3, it could be concluded:

a. As expected the SCOF of the dry tiles is higher than the one of the wet ones, except of one case – Tile 2 measured by group 1T.
b. As expected the SCOF of the water wetted tiles is higher than the one wetted with soaped water.
c. In some cases there are large differences between the SCOF's values obtained by different teams. For example Tile 1 Team 5T obtained SCOF=0.96 while Team 3T obtained the value of 0.57.
d. The standard error for all cases, shown in Table 3, is one order of magnitude lower than the nominal value of the SCOF.

Table 3: Standard error for each test shown in Figures 3.

TILE	TEAM	STANDARD ERROR			TILE	TEAM	STANDARD ERROR		
		DRY	WET	SOAP			DRY	WET	SOAP
1	1R	0.022	0.015	0.027	4	2R	0.022	0.010	0.017
	2T	0.024	0.023	0.013		4R	0.036	0.016	0.017
	3T	0.022	0.025	0.016		6R	0.023	0.023	0.017
	5T	0.041	0.012	0.020		6T	0.053	0.034	0.017
2	2R	0.040	0.013	0.013	5	1R	0.020	0.018	0.016
	2T	0.036	0.018	0.019		1T	0.015	0.013	0.016
	4R	0.037	0.019	0.025		5R	0.032	0.018	0.016
	4T	0.055	0.040	0.033		5T	0.030	0.029	0.016
3	1T	0.016	0.008	0.018	6	3R	0.018	0.016	0.015
	3R	0.018	0.016	0.010		3T	0.026	0.017	0.015
	6R	0.016	0.011	0.016		4T	0.031	0.018	0.015
	6T	0.053	0.014	0.019		5R	0.016	0.022	0.015

IV. EFFECT OF PULLING DIRECTION

The manufacturing process of some tiles is directional, e.g. extruded ceramics tiles, while in others, such as polished marble tile, it is not. To find whether or not the SCOF value is influence by the manufacturing process, t-Test was performed for all pair of pulling directions combinations: NE, NS, NW, ES, EW and SW. The test answers the question if the means of SCOF of the pair are the same or not ($P<0.05$). Table 4 summarized the results obtained by the t-Test: S means that statistically the pair have the same mean and D means that statistically that the means are different.

Although the results of the t-Test indicate that the pulling direction does affect the value of the SCOF in some cases, at least from statistical point of view, one has to realize that practically the differences are small and irrelevant when the SCOF is measured on actual sites where the conditions to perform the test are not ideal. Figure 4 shows the mean values of the SCOF for the four perpendicular direction of the pulls. As can be seen the differences are very small and considering the errors in the testing procedure the errors are not significant.

V. TEAMS' PERFORMANCE

The performance of each team can be measured by the standard errors of their results. Thus, the results given in Table 2 were rearranged and plotted in Figure 10. A small value for the standard error indicates more attention and strict following of instructions (see framed cases in Figure 10 teams 3R, 6R and 1T). On the other hand a large value of the standard error

Table 4: t-Test results.

TILE	CONDITION	DIRECTIONS					
		NE	NS	NE	ES	EW	WS
1	DRY	S	S	S	S	S	S
	WET	S	S	S	S	**D**	**D**
	SOAP	S	S	S	S	S	S
2	DRY	S	S	S	S	S	S
	WET	S	S	**D**	S	**D**	**D**
	SOAP	S	**D**	**D**	S	S	S
3	DRY	S	**D**	S	S	S	**D**
	WET	S	S	S	S	S	S
	SOAP	S	**D**	**D**	**D**	**D**	S
4	DRY	S	S	S	S	S	S
	WET	S	**D**	**D**	S	S	S
	SOAP	S	S	S	S	S	S
5	DRY	S	**D**	S	**D**	S	S
	WET	**D**	S	S	S	S	S
	SOAP	S	S	S	S	S	S
6	DRY	**D**	S	S	**D**	**D**	S
	WET	S	**D**	**D**	**D**	**D**	S
	SOAP	**D**	S	S	S	S	S

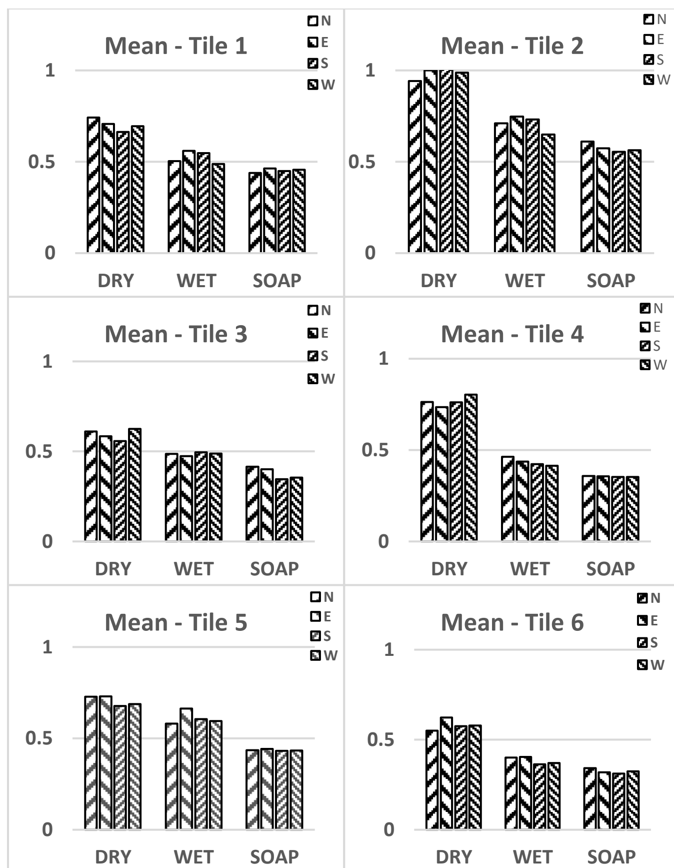
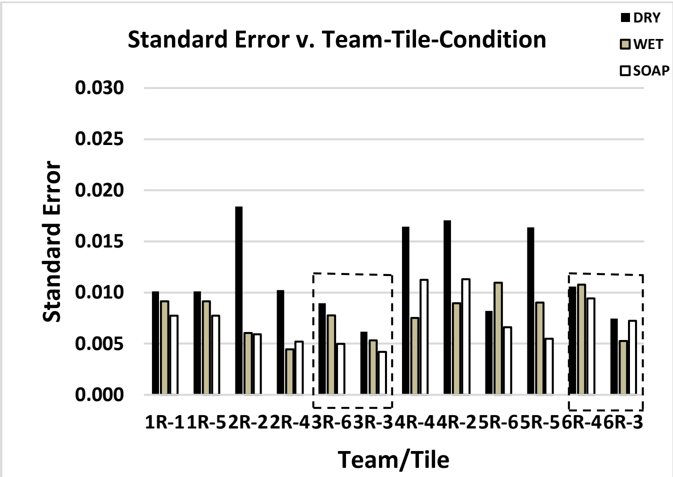


Figure 4: SCOF for all tiles at different pulling directions.

indicates poor attention to the testing procedure (see Figure 10 teams 4T and 6T).



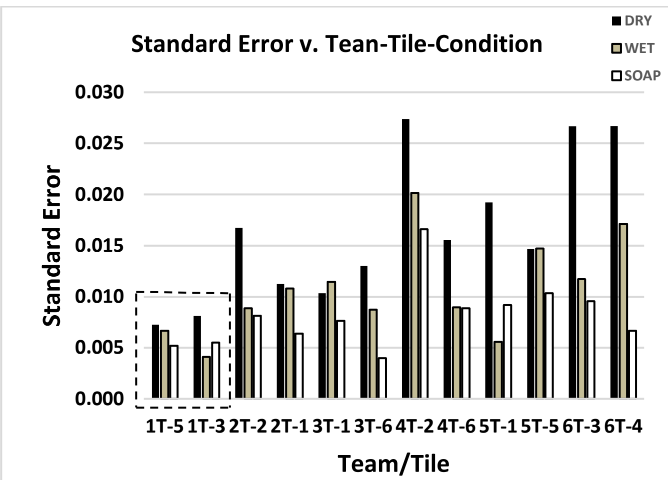


Figure 10: Standard error by team and tiles.

VI. Conclusions

The results of a large number of pull tests that were performed by inexperienced operators on different tile in three different surface conditions have been presented. The results lead to some conclusions:

a. The pull test is valid for wet surfaces given it is performed correctly.
b. Training will improve the results.
c. The use of the right force gauge, which records the peak value of the pulling force, will improve the results.
d. The number of tests cannot be fixed and should increase to the point where the standard error reached a certain value (e.g. 5%).
e.

VII. References

[1] Ian M. Hutchings, "Leonardo da Vinci's Studies of Friction", Wear, 360-361 , 2016, pp. 51-66. http://orthoinfo.aaos.org/topic.cfm?topic=A00130

[2] Tribology History, http://www.tribonet.org/tribology-history/.

[3] Elena Popova, Valentin L. Popov, " The research works of Coulomb and Amontons and generalized laws of friction", Friction, 3(2), 2015, pp. 183-190.

[4] ASTM C 1028-96," Standard Test Method for Determination the Static Coefficient of Friction of Ceramic Tile and Other Like Surfaces by Horizontal Dynamometer Pull-Meter Method.

[5] ANSI/NFSI B101.1 2009, "Test Method for Measuring Wet SCOF of Common Hard-Surface Floor Materials.

[6] ASTM F 609 -96, "Standard Test Method for Using a Horizontal Pull Slip meter (HPS).

[7] "Inherent weaknesses in ASTM C1028 – 07 SCOF Test Method", https://www.c1028.info/weaknesses.

[8] Jungsoo Kim, "Comparison of Three Different Slip Meters under Various Contaminated Conditions", Safety and Health at Work, Vol. 3, No. 1, March 30, 2012.

[9] Kai Way Li, et. al, "Evaluation of two models a slipmeter", Safety Science, Vol. 47, 2009.

[10] Dennis Chimich, et. al.,"Variability of Walkway Tribometer Measurements", https://app.oxfordabstracts.com/stages/123/programme-builder/submission/22562?backHref=/events/123/sessions/52

[11] Wen-Ruey Chang, Simon Matz, "The slip resistance of common footware materials measured with two slipmeters", Applies Ergonomics, Vol. 32, 2001.

[12] A comparison of two slipmeters, ZURICH,Risktopic 4-3.010 October 2012.

[13] Raoul Gronqvist, et. al," Evaluation of three portable floor slipperiness testers", Industrial Ergonomics, Vol. 25, 1999.

[14] Kevin Hallas, Robert Shaw," Evaluation of Kirchberg Rolling Slider and SlipAlert Slip resistance meters", Healt & Safety Laboratories, Harpur Hill, Buxton, Derbyshire, SK17 9JN, 2006

[15] Wen-Ruey, et. el, "Field evaluation of two commonly used slipmeters", Applied Ergonomics, Vol. 34, 2003.

[16] C. M. Powers, et. al,, "Repeatability and Bias of Two Walkway Safety tribometer", Journal of testing and evaluation, ASTM 1999.

[17] Brian C. Grieser, et. el, "Slip Resistance - field measurements using two modern slipmeters", Professional Safety, June 2002.

[18] Raoul Gronqvist, et. al, "The validity and reliability of a portable slip meter for determining floor slipperness during simulated heel strike", Accident Analysis and Prevention, Vol. 35, 2003.

[19] C. M. Powers, et. al,, "Validation of Walkway Tribometers-Establishing a Reference Standard by Powers", Forensic Sciences, Vol. 55, 2010.

[20] Oren Masory, "Comments On ASTM-F2508-13", Journal of Ergonomics, September 30, 2016.

Compliant Joints Actuated Via Cables

Michael Ayer(1), Melissa Morris(2)

(1) Florida Polytechnic University
Department of Mechanical Engineering
Lakeland, FL 33805

(2) Embry-Riddle Aeronautical University
Department of Engineering and Technology
Daytona Beach, FL 32114

morrim44@erau.edu

ABSTRACT

This paper introduces work to investigate the actuation of compliant joints in soft robotics using cable actuators. The general concept is provided. Afterward, the manufacturing of the joints in TPU via 3D printing is investigated with the intent to integrate the cable on the inside. The force-deflection relation of a single degree-of-freedom joint is determined for three different raster angles. The results allow for the consideration of printing angle when producing compliant joints via 3D printing and the corresponding optimal actuator selection.

Keywords

Soft robotics, cable actuation

1. INTRODUCTION

Soft robotics has so far mostly involved the use of silicone formed in molds and actuated by pneumatics. This method is excellent for certain applications and yields a completely soft mechanism. However, there are also problems to be resolved using this material and actuation method including fatigue, risk of puncture, and use in areas with low pressure.

The work presents an alternative actuation method. Cable actuators have been used in robotics for decades [1], [2]. They are a lightweight method of actuation and provide some inherent compliance due to not being rigid. As they are a flexible actuation method, they are a suitable method for soft robot actuation [3]. Typically cable actuation drives rigid links and components, but here the use of soft materials and compliant links are proposed. An example of how this will work is shown in Figure 1. The soft material can be molded silicone or any other flexible material.

As a first tentative step toward efficiently actuating soft components with cable actuation with soft components, this work presents some initial investigation into using 3D printed compliant materials. Using this manufacturing technique offers the possibility of faster prototyping and more versatility in design. It also easily facilitates the routing of cables via allowing for channels in any location and shape. Work was done here to investigate the force-deflection relationships of TPU that has been 3D printed, with consideration of the raster angle of the print.

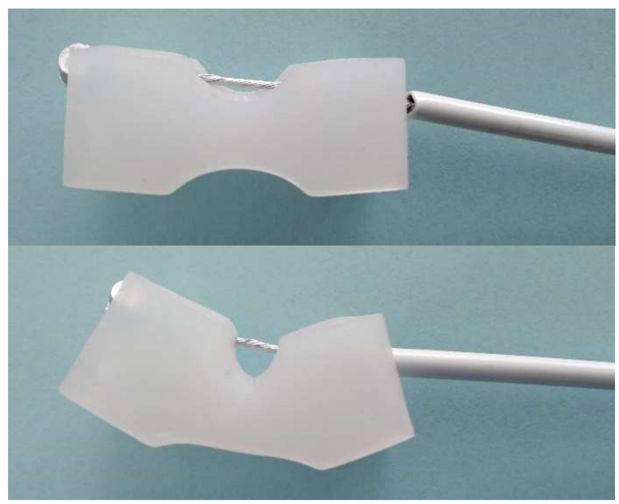


Figure 1. Compliant joint being actuated via cable actuation

This paper begins with some background into compliant links. It then details the ultimate goal of research effort. Next, the details of the experiment for the main objective of this part of the work is described. The paper concludes with the result and discussion of the findings here regarding using 3D printed TPU as the compliant component in a cable-actuated soft robot.

2. BACKGROUND

2.1 Literature Review

Compliant mechanisms are an exciting field that has many advantages. The ability to flex comes from innovative materials and manufacturing techniques. The past few years has introduced compliant materials that are used as filaments for 3D printers. This facilitates rapid manufacturing of diverse shapes. However, there are still issues that hinder the adoption of printed compliant mechanisms. Traditional mechanisms are designed using materials with known mechanical properties that can be modeled and optimized using Finite Element Analysis (FEA). Unfortunately, this method gives large errors when used design a compliant 3D printed joint. This is because material properties are not well known and studied for this instance. Instead, the design and implementation of 3D printed compliant parts is mostly done via trial-and-error.

Material considerations are a major issue in compliant joints. Many examples can be found in prosthetic and anthropomorphic hands. Work on an anthropomorphic hand utilizing rubber joints concluded that further research is needed in the material and manufacturing method to increase durability [4]. Another study designed and evaluated a soft prosthetic hand that noted that additive manufacturing methods and technologies have increased the design possibilities [5]. They observed changes in stiffness due to variations in infill density of 3D printed parts. In recent work, an endoskeleton was 3D printed for an anthropomorphic hand [6]. Compliant joints were used with the endoskeleton and different materials were considered with FEA analysis. However, the designed joint was not able to replicated using 3D printing.

Underactuated mechanisms combine compliant joints and cables. They have more degrees of freedom than they have actuators, allowing for reduced costs and increased reliability [7]. For use in a hand, slack cable-drives in an underactuated condition have investigated [8].

Rapid prototyping is being exploited for designing compliant mechanisms. Most commonly used 3D printers use fused deposition modeling (FDM) [9]. This process leaves behind small gaps, or microchannels, between the material that is extruded. When a compliant part is bent, the gaps become a place for material to go. The orientation of the extrusion, or raster angle, affects the ease of bending at a given angle. Therefore, the mechanical properties of 3D printed parts are very anisotropic and are not well suited for using design methods such as FEA [10]..

Compliant mechanisms are more difficult to design than rigid mechanisms [11], and as noted above 3D printing complicates it further. More research is needed to characterize the mechanical properties of compliant materials and how the properties change with varying manufacturing techniques. In addition, standards need to be developed to verify these mechanical properties. In the meantime, designers need to prove the structural and mechanical properties of their designs.

2.2 Overview of Work

The final goal of this project is to eventually develop the means to efficiently design cable-driven soft robotics. There are many factors that will be investigated and incorporated into the models in order to complete the goal. These include the placement and routing of the cables, cable selection, optional sheathing, the shape of the joint, the size of the joint, and the material in the joint. All of this can then be used to determine the sizing of actuators to meet the goal of optimal actuation. In addition, the manufacturing method needs to be determined, which will affect many of the other aspects.

While the use of silicone or rubber formed via molds is certainly possible and was used for initial concept testing, 3D printing allows for more versatility in the cable channel routing. A channel of any shape can be made in a compliant link made all at once. The printing process can halt at key points for the cable to be placed within the joint and then continue, perfectly embedding the cable in the desired path within the joint. 3D printing is also becoming more accessible and can be done easily without the need for measurement devices nor vacuum chambers to remove bubbles from molded components.

This paper presents findings from the first phase of the planned work. To start, some initial measurements of the material properties of a sample joint were tested. The objective here was to investigate using compliant 3D printed 1-DOF joints with cable actuation. Specifically, the raster orientation and force-deflection relationship is of interest as it is known that this affects the properties of the joint [12].

3. JOINT TESTING

This phase of the project involves investigating the force-deflection relationship of a compliant joint. The raster angle effects in a compliant joint are noted as well. This was done for a simple 1-DOF joint with semi-circular cutouts to create a hinge. The shape was selected for simplicity in order to gain some initial data. A later phase of this work will look at different joint shapes and design and the process used here will be repeated for other shapes.

Due to soft robotics having a lot of promise for offering passive safety when working alongside humans, there is use for robotic manipulators that are human-sized. Therefore, hinge joints with roughly the same dimensions of a human finger were selected for this initial testing project. These joints were printed in TPU with varying raster orientations. The different raster orientations that were tested are: 0°, 45°, and 90° as depicted in Figure 2. The raster angles were produced during the slicing process when the CAD model was being prepared for the 3D printer and are the three most likely to be selected by most users who would chose to manufacture a soft robot joint using this method.

For finding the force-deflection relationships, a means was constructed to cause the actuation with a cable as shown in Figure 3. The data was captured manually from observation and images. Images were captured from a camera mounted a tripod. A protractor was used to measure the angle. A cable was draped over the test hinge with weights hung upon the end to create the applied force. Each specimen was secured via a clamp and carefully aligned. For the testing process, load was kept on the joint as slack cables do not allow for controllable movement. However, the no-load angle of the joint was noted because there was some initial bending due it its own weight.

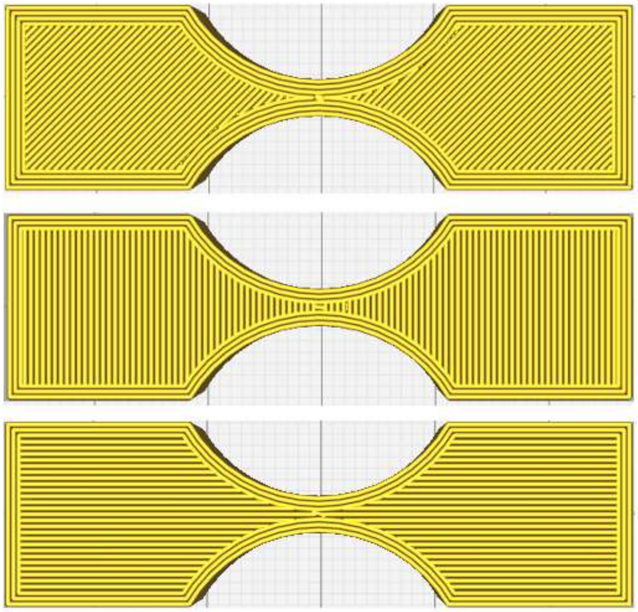


Figure 2. Raster angles at 45° (top), 90° (middle), and 0° (bottom)

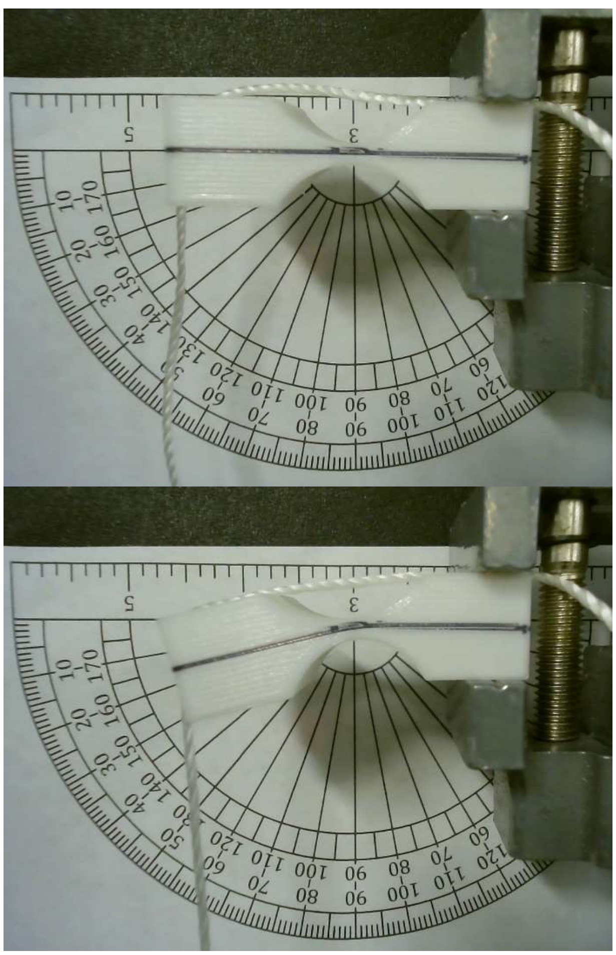


Figure 3. Specimen testing. Unloaded (top) and with a load (bottom)

4. RESULTS AND DISCUSSION

As expected based on findings from literature, there was a difference in the deflections due to the raster angle for compliant 3D printed parts.

The applied force was increased starting from .4905N (50g) to 1.4715N (150g). Figure 4 summarizes the force vs the deflection for all three raster orientations.

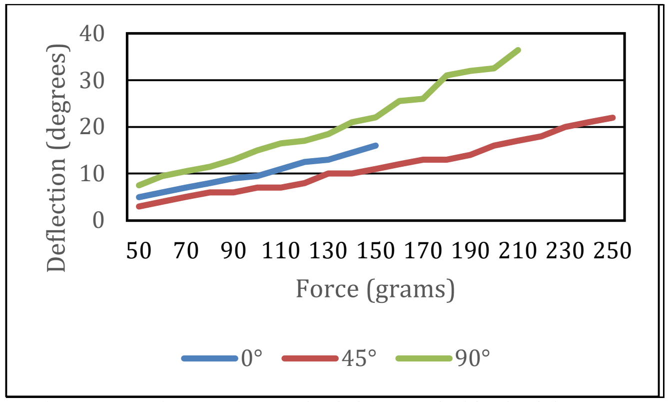


Figure 4. Specimen testing. Unloaded (top) and with a load (bottom)

Initial analysis of the data also shows a decreased angular deflection with hinges printed with a 45° raster orientation compared to 0° and especially 90° raster orientations. This was as expected from other studies of rigid links. The 45° raster orientation inherently supports the hinge against bending via the manner an angled truss supports a horizontal beam. Likewise, the 0° raster orientation provides some support because the force is orthogonal from the layers. The 90° raster orientation puts the microchannels in a position to easily accept the deformation of the printed material and so the hinge bends more easily.

One clear unintentional result found was that the plastic deformation limit was significantly influenced by the raster angle. For the 0° raster orientation, elastic deformation occurred at 1.4175N. This was much sooner than at the other orientations as shown in Figure 4. The printing layers being parallel with the hinge do cause significant weakening of the hinge. This is not entirely unexpected given that the layering position would cause significant tension on the top layer and compression on the bottom layer of the hinge. These layers do not have further supports as is inherent with the other raster angles.

5. CONCLUSION

This paper shows that raster angle of compliant 3D printed hinges does affect the force-deflection relationship. Therefore, it is important to consider this when utilizing this method of manufacturing in soft robotics applications.

In the future, additional tests will be conducted on raster orientations to complete this phase of the project. More detailed results will be obtained via the use of digital image correlation in order to obtain data on the stress and strain.

Further, electronic actuation of the cable will be implemented so that cyclic testing may be performed on the hinges as well. Eventually, multiple-DOF mechanisms will be investigated so that a tool for optimally designing cable-actuated soft robotics can be produced. This will conclude after all phases of the project investigating other aspects of a cable-actuated compliant link.

6. ACKNOWLEDGMENTS

We wish to thank the help of Edward Shynski for his help in preparing some of the experimental parts of this work.

7. REFERENCES

[1] J. S. Albus, R. V. Bolstelman, and N.G. Dagalakis, "The NIST robocrane," *J. Robotics System*, vol. 10, no.5, pp. 373-385, Sept. 1992.

[2] L. W. Tsai, "Design of tendon-driven manipulators," *J. Mechanical Design*, vol. 117, no. B, pp. 80–86, 1995, doi: 10.1115/1.2836474.

[3] M. Borgatti and K. Love, "Cable pull mechanisms – ready, willing, and cable," *Make: Soft Robotics*. San Francisco, CA: Maker Media, 2018, ch. 4, pp. 37-56.

[4] T. V. J. Tarvainen, W. Yu, and J. Gonzalez, "Development of MorphHand: design of an underactuated anthropomorphic rubber finger for a prosthetic hand using compliant joints," *IEEE Int. Conf. ROBIO*, 2012, doi: 10.1109/robio.2012.6490957.

[5] I. Hussain, Z. Iqbal, M. Malvezzi, L. Seneviratne, D. Gan, and D. Prattichizzo, "Modeling and prototyping of a soft prosthetic hand exploiting joint compliance and modularity," *IEEE Int. Conf. ROBIO*, 2018, doi: 10.1109/robio.2018.8665231.

[6] M. Tavakoli, A. Sayuk, J. Lourenço, and P. Neto, "Anthropomorphic finger for grasping applications: 3D printed endoskeleton in a soft skin," *Int. J. Adv. Manufacturing Tech.*, vol. 91, no. 5-8, 2017, pp. 2607–2620. doi: 10.1007/s00170-016-9971-8.

[7] R. Ozawa, K. Hashirii, and H. Kobayashi, "Design and control of underactuated tendon-driven mechanisms," *IEEE Int. Conf. Robotics and Automation*, 2009, doi: 10.1109/robot.2009.5152222.

[8] S. Shirafuji, S. Ikemoto, and K. Hosoda, "Design of an anthropomorphic tendon-driven robotic finger," *IEEE Int. Conf. ROBIO*, 2012, doi: 10.1109/robio.2012.6490995.

[9] D. T. Pham and R. S. Gault, "Solid ideas [rapid prototyping]," *Manufacturing Eng.*, vol. *75*, no. 5, pp. 239–243, Oct. 1996. doi: 10.1049/me:19960514.

[10] J. T. Cantrell, S. Rohde, D. Damiani, r. Gurnani, L. Disandro, J. Anton, and P. G. Ifju, P. G. "Experimental characterization of the mechanical properties of 3D-printed ABS and polycarbonate parts," *Rapid Prototyping J.*, vol. *23* no. 4, pp. 811–824, 2017, doi: 10.1108/rpj-03-2016-0042.

[11] L. Howell, *Compliant mechanisms*. New York: Wiley, 2002.

[12] P. Messimer, B. O'Toole, and M. Trabia, "Identification of the mechanical characteristics of 3D printed Ninjaflex," *Proceedings of the ASME Int. Mechanical Engineering Congress and Expo.*, 2019 , IMECE2019-11674.

Trained Probabilistic Models for the NAO Robot in a Labyrinth

Samantha Eaton, Kleanthis Zisis Tegos, Sagar Vilas Jagtap, Akram Alghanmi, Godwyll Aikins, Neda Keivandarian, Ran Bi, Jesse Torres, Kholud Alghamdi, Marius Silaghi

Florida Institute of Technology

ABSTRACT

A framework based on a Probabilistic Model for the moving behavior of the NAO humanoid robot in the environment given by $20in \times 20in$ vinyl maze cells is being trained and made available for future applications. NAO is one of the most advanced humanoid robots, having advanced speech, vision, and behavior based on artificial intelligence already implemented on it, and being a precursory of the larger Pepper robot famous for being used as host at certain hotels. Pepper uses wheels, probably since the leg-based movement of Nao proved hard to harness with precision and robustness. Indeed, most of the intelligence currently present in NAO is speech and gesture-related, while its autonomous walking capabilities are only little exploited in existing software, and only for simple behaviors, such as following a red ball.

We test that it is possible to exploit a public NAO sensor database made recently available, to build a sample probabilistic model for walking and turning in a controlled vinyl maze. The probabilistic model is made of a new and more general random variable-based representation of phenomena and with conditional probability tables for the NAO sensors given relevant environment states allowing for complex planners and reasoners, that are based on rich POMDP models, to be built on top of it. The proposed model is tested with a simple particle filter localizer on a predefined trajectory, and improvements and data missing in the NAO database are being identified for future work.
Keywords: Humanoid, framework, model, AI, Nao.

1. Introduction

The humanoid NAO robot, together with its wheeled larger version Pepper, are among the most advanced robots available to the general public. These robots are known to gracefully dance, walk, talk and entertain the users with intelligent dialog. However, the walking intelligence of this robots is rather low, and no common application enables NAO to autonomously and robustly walk; not even in restricted environments.

In previous work, groups of researchers in our lab started to address this problem by standardizing a restricted maze environment based on $20in \times 20in$ white vynil cells [1], Figure 1.a,b, and started organizing a public database publishing sets of contributed sensor data and movement measurements [2], usable for characterizing and training models of the interaction between Nao and such restricted environments. However, the test of the Nao sensor database was very preliminary and in this work we undertake the extension of the data gathering and framework formalization, as well as testing for the Nao robot.

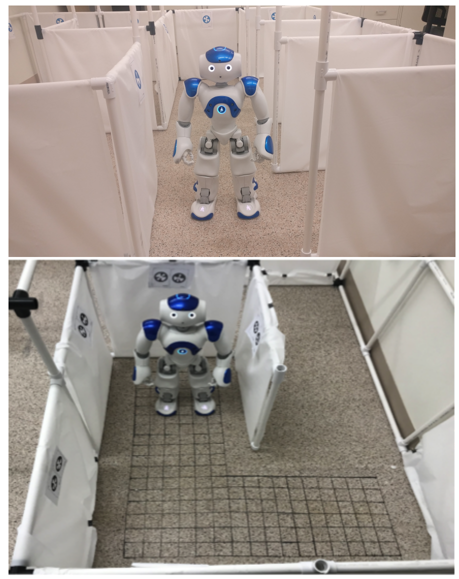


Figure 1. (a) Restricted Environment; (b) Markers for Standardizing Data Measurement

We propose a more general probabilistic model of the movement capabilities of Nao in a controlled environment, defined by the labyrinth of 20 inch square rooms with vinyl white walls.

2. Background and Related Work

Let us introduce the main background on dynamic Bayesian networks and on Partially Observable Markov Decision Problems (POMDPs), which are among the most principled general frameworks for addressing robotic problems.

2.1 Bayesian Networks

A Bayesian network is a graphical probabilistic model representing conditional independence information with graphs consisting of nodes and edges. A node represents a random variable, and an edge represents a direct conditional dependency among the random variables (nodes) it links. Edges are directed with parents preferably being selected as causes of descendants, which simplifies estimation of conditional probabilities. Nodes are conditionally independent of indirect ancestors in the graph given certain nodes in-between (a Markov blanket). The domains of the random variables could be either discrete or continuous. The random variables in the Bayesian networks can be classified in three types: observable (evidence), non-observable (hidden), and controllable (also evidence, graph source nodes). Generally graphs in Bayesian network are constructed to be acyclic [3]. Every node is associated with a conditional probability table specifying its conditional probability given all possible value combinations for its parent variables in the graph, or some compressed representation thereof.

A Bayesian Network is a compressed representation of the full joint probability distribution of its variables. The main application of using Bayesian networks is the enabling of probabilistic inferences in environments where state or phenomena are uncertain or non-deterministic.

A Dynamic Bayesian Network (DBN) is a Bayesian network of a special type, where sets of nodes are repetitions of a pattern in a sequence that can potentially be infinite. It is frequently used to represent sequential processes, each repetition corresponding to a time step, where the time is discretized. It thus represents a probabilistic temporal model. Dynamic Bayesian networks are generative models. The graph in DBN is composed of a sequence of time steps or slices, namely $(0,...,t-1,t,t+1,...)$ [4].

In more detail, Particle Filtering, one of the main reasoning techniques for filtering in DBNs, works as follows. After an initial population at time t with N samples from the hidden state variables set x_t is created from the distribution $P(X_t)$, firstly, for each subsequent populations sample, its next state value x_{t+1} is generated based on the transition model $P(X_{t+1}|x_t)$ and then weighted by the likelihood that it assigns to the corresponding new evidence e_{t+1}, $P(e_{t+1}|x_{t+1})$. Lastly, this weighted population is interpreted as a probability distribution $P(X_{t+1})$, that in a subsequent round is resampled per the aforementioned procedure to generate the next set of samples, thus repeating these inference operations at each remaining time step.

2.2 POMDPs

The Partially Observable Markov Decision Process (POMDP) is a framework for modeling dynamic uncertain environments, adding rewards associated to states and/or transitions in Dynamic Bayesian Networks. A POMDP is defined by a set of states, a set of possible observations, a set of actions, a transition probability function between state repetitions, observation conditional probabilities, an initial belief, a discount factor, and rewards associated to states.

Through observation of evidence, the robot will update the probability distribution of its current state in order to optimize its choice of the next action maximizing future rewards. Actions at a time step contribute in the next state transition.

A reward represents the benefit of a state (and potentially, action). The typical goal of reasoning in a POMDP is to generate an optimal policy (map belief-action) which maximizes the amount of future rewards [5]. A belief is a probability distribution over states. The robot will optimize the next action, making decisions, based on its current belief [6].

2.3 The Nao database structure

We use the database of experimental measurements constructed by the previous work in [2]. This database contains sensor and transition measurement data, as well as its documentation and relevant code snippets. The sensor data contains the measurements obtained from the various sensors on the NAO robot; the left and right sonar sensors for detecting objects and the visual sensor for detecting the standardized landmarks on the walls, see Figure 2. The transition data is measured as the center of mass displacement when the robot is given the command to move between two adjacent squares, each $2in \times 2in$, in the labyrinth.

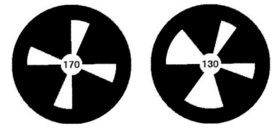


Figure 2. Standardized wall markers, in pairs

The transition data is organized into 4 column datasheets. Each datasheet describes experiments with a given action, and its columns are: the given position, given orientation, recorded position, and recorded orientation. As the name of the columns suggests, each datasheet records actual responses measured for the command associated with it.

The sensor data consists of 12 column datasheets.

- Two columns were dedicated for the position of the robot in the cell of the labyrinth, namely the "Row" and "Column" in the maze.
- "Angle" is the degree of the rotation of robot with respect to the reference vector.
- "Head Pitch" and "Head Yaw" (the orientation of the robot's head) are represented in the sensor datasheets but were controlled at the still position and not used in the localization of the robot.
- The "Left" and "Right" sonar sensor readings of the robot were the most robust Sensor data readings.
- In the Nao sensors database data, two landmarks are placed on each maze wall center, and data pertaining to their detection was recorded in the sensor data: "Landmark detection" (whether the landmark was detected), "Landmark ID" (Number ID on the Nao Mark), "Alpha" and "Beta" (which are the line of sight angle of the landmark's position from the robot's head axis), and "Width" and "Height" of the detected landmark.

2.4 Related Work

In [7], a Nao robot is addressing the problem of simultaneous localization and mapping (SLAM) in a room which has multiple Nao marks with random locations. The contribution proposed to enhance SLAM addressing the real-time incorporation of new landmarks in exploration.

An autonomous algorithm to obtain the parameters of the POMDP in a navigation system for indoor environment based on WiFi and ultrasound observations is implemented in [8]. This algorithm can learn the observation and transition matrix in autonomous mode which is coming from wifi simultaneous localization and mapping so it could obtain the WiFi and Ultrasound environment map with minimum effort. The localization algorithm converges faster by adding a global navigation system sensor.

A multimodal interaction system between a human and a robot was suggested in [9] by expanding a POMDP framework in a humanoid robot called "Pepper" to observe various multimodal information streams with its own sensors. The interaction system is made of two strata in the framework of POMDP – set the first stratum so as it decides to start interaction based on the physical

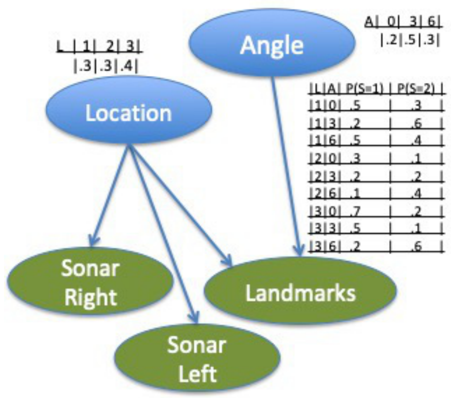


Figure 3. Conceptual Belief Network for Observing the Situation

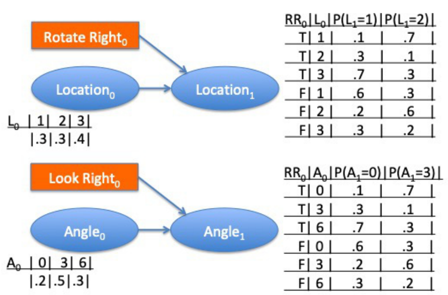


Figure 4. Conceptual Belief Network for Transitions

distance between Pepper and a user, and set the second stratum to control multimodal interaction. The effort led to stratified interaction to reduce the increase of the user's belief states, making the problem more manageable.

A variable resolution technique is proposed in [10] for reducing the complexity/state space of the POMDP. This is accomplished by automatically adjusting the number of the states in the grid based on features of the environment map while maintaining the level of detail required for planning at a given region to find a robust and efficient policy. This is validated in a POMDP-based simulation of a navigation environment, a realistic robot simulator, and an autonomous robot.

3. Concepts and Data Structures

To model the phenomena taking place when a Nao robot is observing or walking in the maze, belief networks are designed, as

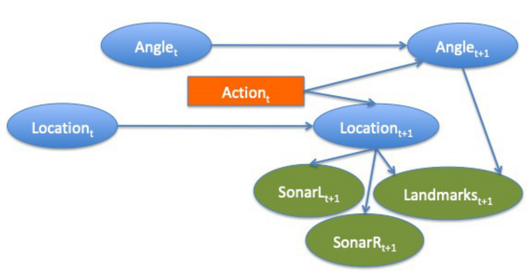


Figure 5. Dynamic Belief Network utilized

conceptually shown in Figures 3 and 4. The Bayesian Network in Figure 3 abstracts the relations between: the relative position or "Location" of the center of the robot (the center of the space between its feet) with respect to walls, the torso heading angle, the angle between its head and torso, the measurements of its sonars, and the landmark detection (the actual sets of values of the variables are more complex, and described later).

The Bayesian Network in Figure 4 describes abstractly the impact of actions on location and angles, for rotations. Similarly, the actual sets of values of the variables are more complex, and described later. With most types of walking, both the previous angle and previous location have an impact on the resulting location.

The Location or position node holds one of the M unique $x-y$ coordinate points within the labyrinth, and possible heading angles of the torso. If the maze would have m rows and n columns, them $M = m \times n$. The Angle node represents the angle between the head and torso, as controlled by the robot.

The Left Sonar node represents the measured value of the left sonar based on the $x-y$ coordinate point within the labyrinth and the overall angle the robot body is facing. Its conditional probability table (CPT) is a table of size $M \times N \times O$, where M represents the total number of unique $x-y$ coordinate points within the labyrinth, N is the number of unique torso heading angles, and O is the number of possible sonar readings. The Right Sonar node is defined similarly, but represents the measured right sonar values.

The landmark node represents the event of detecting some of the landmarks in the environment. It is related to the $x-y$ coordinate position within the labyrinth and the overall position angle of the head of the robot. Its conditional probability table (CPT) is compressed based on an independent success assumption into a table of size $M \times N \times L$, where M represents total number of unique $x-y$ coordinate points within the labyrinth, N is the number of possible unique sums between torso heading angle and the torso-head angle, and L is the number of possible landmarks.

A possible value is any bag of landmarks with their possible sizes and positions in Nao's visual field. The representation can be compact by using a single Boolean per landmark to represent it's visibility, while the distribution of the rest of the landmark location and size values can be integrated as deterministic nodes with Gaussian noise.

This representation is more general and flexible than the earlier version [2] where there are three landmark nodes per time frame, that can recognize up to maximum three landmarks, an assumption that is not always true with complex mazes. The bag representation also simplifies the handling of matching different landmark IDs in different locations. The case of landmarks with repeating IDs is described later.

While the shown figures display conditional probability tables for illustration, the actual conditional probability tables of these Bayesian networks are estimated by counting from the measurements in the Nao sensor database, as described further in the article.

4. Algorithms

The database extension we provide offers algorithms for estimating the belief (i.e., conditional probability distribution), given current sensor measurements, as well as for filtering the new beliefs along sequences of actions and measurements of sensor readings .

4.1 Training the Conditional Probability Tables

Building on top of the sensor data made publicly available by a previous work [2], libraries for estimating the conditional probability tables for our Belief Network (refer to Figure 3) were developed.

The Sonar nodes each relate data from the respective sonar sensors with the location and angle information. Extracting the sonar data for each sonar results in a $(m \times n) \times N \times O$ matrix. Where N is the number of torso heading angles recorded at each position and O the number of sonar readings handled. As some measurement readings never occurred in the training set leading to zeroes as shown below, the probability mass is later redistributed to compensate for possible noise by assigning a small probability to each possible value.

Sonar Left Node:

X	Y	Angle	P(Sonar=0.46)	P(Sonar=0.47)	...
0	0	0	0.339	0.000	...
0	0	11.25	0.000	0.000	...
...	...	...	...	...	...
m	n	N	...	...	...

For the Landmark node, we separately represent the Boolean landmark detection event and its actual parameters in terms of position. When a landmark is detected its position is pretty accurate and therefore can be approximated with a deterministic function with small Gaussian noise. For the Landmark detection event, the CPT can be represented compactly by exploiting an *assumption of cause independence between individual landmarks' detection success*. As such one only needs to store separately the probability of detecting each individual landmark, and the probability of each bag of landmarks can be inferred from these assumption of independence with the product rule. The resulting table is a $(m \times n) \times N \times L$ matrix. Where L is the number of landmarks in the environment. At training the landmark data is True if the landmark was encountered, and False otherwise.

Landmark Node:

X	Y	Angle	Landmark 1	Landmark 2	...
0	0	0	FALSE	FALSE	...
0	0	11.25	FALSE	FALSE	...
...	...	...	...	...	...
m	n	N	...	...	...

For the case where landmarks with the same ID are reused in the environment, the compact representation needs to include one column for each count of detected markers with identical IDs, and the aforementioned deterministic probability node linking the position to the parameters of the detection are normalized over all such repeated occurrences.

Transition probabilities for the NAO humanoid robot were calculated using the transition data that came with the dataset. Recorded readings and each action command sent to the robot were used to compute the probabilities by counting.

$$P(r|0,S) = \frac{\text{Number of times state r = 0}}{\text{Total times the command was executed}} \quad (1)$$

From the Nao database, the following translational conditional transition probability table of size $l \times o \times r$ was produced. Where l is the relative state the robot was asked to move to, o is the relative orientation the robot was asked to be in, and r is the possible state the robot may be in after execution of a command. Both $l, r \in S\{0,1,2,3,4,5,6,7,8\}$, in which each number represents either the robot's current or neighboring position.

Translational Transition Probability for Location Node:

$State_{cmd}$	$Orientation_{cmd}$	0	1	...	8
0	-169.75	1	0	...	0
0	-158.50	0.1	0.4	...	0
...	...	...	...	...	...
8	180	...	...	...	...

A similar rotational conditional transition probability table is built for all possible resulting rotations as a result of the action. The rotation and translation deviations are considered to be independent variables.

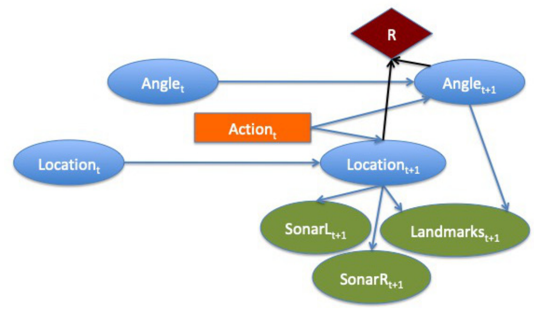


Figure 6. Dynamic Decision Network

4.2 Particle filtering

From the Bayesian Network concept describing observation and transition phenomena in Figures 3 and 4, we build a Dynamic Belief Network shown in Figure 5.

Performing exact inference for Dynamic Bayesian Networks (DBN) is hard for complex networks, given that the cost of the update procedure is $O(d^{|x|+2})$ for domain sizes d with $|x|$ variables in a densely connected graph, as stressed in [3]. Particle Filters described earlier yield an efficient family of algorithms for inference using sampling for DBNs [3]. The main principle of this approach is that the high-probability regions of the state space are to be highlighted by representative sampling of a new generation of samples at each DBN time step as starting point for the weighted sample space of the next time step. This is filtered to the subsequent time step by computing a weight value for every sample from evidence. At the next time step one is again generating with higher probability new samples in the neighborhood of the samples with the highest weight.

For this sample generation, two approaches are proposed. In the first approach, the distribution used in sampling comes from a pooled multivariate Gaussian mixture with weights given by the previous round samples. Practically, the distribution used for resampling is interpolated using a radial-basis function with Gaussian kernel. In the competing approach, the distribution used for sampling is obtained with multi-variate interpolation (Shepard method or linear) [11].

The particle filtering infers the location of the robot within the cell, given its past evidence locations. Integrating rewards conveying the utility of goal states, a dynamic decision network is obtained, as in Figure 6.

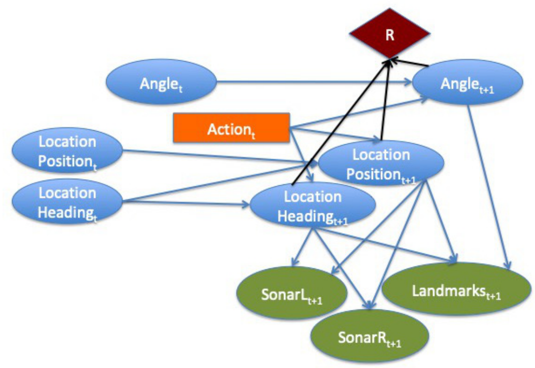


Figure 7. Dynamic Decision Network optimized

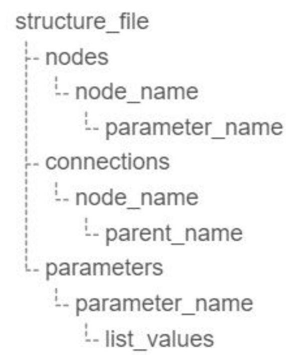


Figure 8. Layout of JSON file containing Bayesian Network

Since for many actions of the humanoid it can be assumed that the transition is independent for location heading and location position, the CPT storage space is improved with the finer grain dynamic decision network in Figure 7, where the "Location" random variable in the previous models is replaced by two random variables: "Location Position" and "Location Heading", for the $x-y$ coordinate and for the torso heading, respectively.

5. Experiments

Preliminary experiments that enabled extensions to the NAO sensor database with additional landmark measurements were achieved, despite the 2020 social quarantine which made safe team-based handling of Dougy, our Nao v5 robot, particularly hard.

5.1 Setup

As shown in Figure 1b, the humanoid was placed at the starting point with three immediate landmark pairs around it (left, right, and back). Markers were also drawn on the floor beneath the humanoid to standardize the observation of the change in position results.

A series of trials was executed using the humanoid to find the left and right sonar readings and the landmark visual readings based on the various $x-y$ coordinate points within the labyrinth and the torso and head positions. The generated data was saved in CSV format, added to the database, and used to generate the belief network.

5.2 Obtained Models

The belief network with sample node data is shown in Figure 3. The main inputs to the overall system is the $x-y$ position coordinate point and the angle of the head.

In our experiments, M is 32 for the unique $x-y$ coordinate points within the labyrinth and N is 32 for the unique degree angles (0 degrees to 360 degrees in steps of 11.25 degrees). Data in the database was available for $L=3$ distinct landmarks.

5.3 Model Guide

The goal of this work was to provide baseline models upon which the locational awareness of the NAO robot can be improved. This section serves as a reference on how the obtained models, which are made publicly available in the Nao database, can be used.

```
BN estimate_CPTs(BN_structure, training_data, BN)
```

The Python function `estimate_CPTs` calculates the conditional probability structures, given the Bayesian network structure and the corresponding training_data measurements, which are stored in a file. It accepts three file names, with the BN_structure and output BN files having JSON format, and the measurements having csv format. The format of the BN_structure is shown in Figure 8.

The requirement for the measurements file is that the first row contains the column names, which correspond to the nodes of the network. The output file only stores the matrices, and no structure information regarding the network, since that can be extracted from the structure file.

```
distrib belief(l_sonar, r_sonar, landmarks, theta)
```

The function `belief` is used to compute a probability distribution for all the possible current locations (x,y,α) within a cell given the sonar and landmark readings, as well as the head-torso angle `theta`.

```
distrib likelihood_weighting(DBN, query_vars,
                evidence_vars, samples_nb)
```

The likelihood weighting function calculates the probability distribution over the query variables given the evidence variables. It accepts the DBN as a filename for conditional probability tables and for its structure, the query variables as a dictionary with keys being the needed node names, the evidence as a dictionary with keys being the node names and values being the measured node state, and a number of samples to be run. The more samples are run, the higher the accuracy of the likelihood weighting algorithm.

API of the Bayesian Network.
The Bayesian Network was developed using Python to parse the data results from the CSV files and generate the expected node results. The generated conditional probability tables (CPTs) are written to CSV files.

API of Likelihood Weighting Module.
The Likelihood Weighing Module was developed using Python. The code parsed the Bayesian Network conditional probability tables (CPTs), the user input specifying the positional goals of the robot the expected current position of the robot, and the robot's sonar and landmark detection data based on it's current position. The code produced the transitional probabilities for the next timestep at that location, and the estimated final robotic location with head orientation.

6. Future Work

The impact of the new model on the accuracy of the prediction on the position of Dougy, our Nao robot, is being evaluated but this effort is being delayed by the quarantine imposed on the robotic equipment by the state of Florida. We hope to have such accuracy results available for the next version of the paper. Other future experiments consist in extending the built models for more complex applications.

7. Conclusions

The problem of lack of support for easy development of intelligent and autonomous Nao robot applications with walking mobility is addressed by extending on the preliminary prior work in our organization, that introduced a "sensors and transition measurements database" for probabilistic reasoning in restricted environments.

This work extends prior contributions not only by adding new data measurements with new sensor and landmark sensor measurements, but also by contributing a more general sensor model, and a Dynamic Decision Network framework with libraries for training conditional probability tables and for particle filtering inference steps compatible with the data formats in the Nao sensors and transition database.

While the amount of the contribution was limited and kept preliminary by the limits on team work put by the unprecedented challenges of the crisis of the year 2020, it nevertheless helps clarify ideas, improve models, and provides a step forward in the effort for better support of intelligent, mobile robotics with Nao humanoids.

References

[1] A. Hasanain, T. Weekes, M. Person, K. Paul, Y. Chang, A. Rothman, Z. Rui, R. Rahil, M. Syed, C. Wodd, and M. Silaghi, "Models and framework for supporting humanoid robot planning & exploration," in *FCRAR*, 2018.

[2] K. Alghamdi, J. Torres, J. Burden, V. Maddineni, R. Alharbi, T. Neelam, P. Tarsoly, A. AlHaif, J. Nke, S. Jadhav, S. Dodle, and M. Silaghi, "Towards sensor and motion measurements databases for training models for the nao robot," in *FCRAR*, 2019.

[3] S. J. Russell and P. Norvig, *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[4] K. P. Murphy, *Dynamic Bayesian Networks: representation, inference and learning*. PhD thesis, University of California, Berkeley, 2002.

[5] J. Luo, S. Zhang, X. Dong, and H. Yang, "Designing states, actions, and rewards for using pomdp in session search," in *Advances in Information Retrieval* (A. Hanbury, G. Kazai, A. Rauber, and N. Fuhr, eds.), pp. 526–537, Springer International Publishing, 2015.

[6] N. Roy, G. Gordon, and S. Thrun, "Finding approximate pomdp solutions through belief compression," *Journal of Artificial Intelligence Research*, vol. 23, p. 1–40, Jan 2005.

[7] Y. Zhang, *Real-time SLAM for Humanoid Robot Navigation Using Augmented Reality*. PhD thesis, Applied Sciences: School of Mechatronic Systems Engineering, 2014.

[8] M. Ocana, L. M. Bergasa, M. A. Sotelo, and R. Flores, "Indoor robot navigation using a pomdp based on wifi and ultrasound observations," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2592–2597, Aug 2005.

[9] S. Iijima and I. Kobayashi, "A pomdp-based multimodal interaction system using a humanoid robot," in *Proceedings of the 30th Pacific Asia Conference on Language, Information and Computation: Posters*, pp. 519–525, 2016.

[10] R. Kaplow, A. Atrash, and J. Pineau, "Variable resolution decomposition for robotic navigation under a pomdp framework," in *2010 IEEE International Conference on Robotics and Automation*, pp. 369–376, 2010.

[11] D. Shepard, "A two-dimensional interpolation function for irregularly-spaced data," in *Proceedings of the 1968 23rd ACM national conference*, pp. 517–524, 1968.

A Neurofeedback-Driven Humanoid to support Deep Work

Troy Weekes and Thomas C Eskridge

L3Harris Institute for Assured Information
Learning, Abstraction, Similarity and Reasoning (LASR) Lab
Department of Human Centered Design
Florida Institute of Technology

ABSTRACT

High performance is desired in the workplace, even with swarms of robots on their way in the Fourth Industrial Revolution. Our research focuses on the population of knowledge workers, who are typically expected to sit in one space for extended periods while performing deep, intellectual and creative work. For those who work predominantly using computers, there is growing scope to augment task performance using artificial virtual agents. This trend is evident in the adoption of voice-based, and gesture-based applications that allow users to issue vocal or gestural commands while their hands are occupied on primary tasks. Even though multimodal interaction may yield more productivity than solely mouse and keyboard interactions, it may still impose a significant cognitive load on the user. We propose the modeling of a smart motivational humanoid assistant that is personalized to interact with human users without explicit commands, and instead via wireless sensors that can perceive the operator's brain activity. The humanoid engages with the human using effective nudges through neurofeedback.

Keywords: Robotics, Neurofeedback, Deep Work

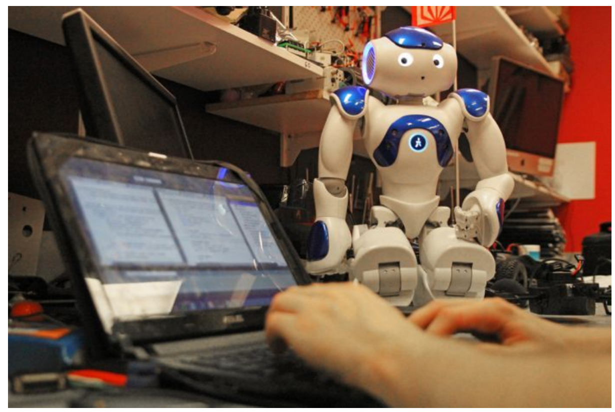


Figure 1. Proposed context of use shows Nao humanoid interacting with a knowledge worker that wears a Muse neurofeedback headband while completing work tasks using a computer in the office workspace. Image courtesy ABC Gold Coast: Damien Larkins.

1. Introduction

Knowledge workers perform "non-routine cognitive jobs" whose ideas and concepts are responsible for the growth and development of businesses around the world. In the US, knowledge workers constitute almost 50% of the workforce [1]. They work with tools and automation that relieve them of many tasks, enabling them to deeply immerse themselves into the difficult knowledge tasks that cannot be automated [2]. The research question arises, how should human workers interact with complex automated technologies, which is itself a knowledge-work task? And, can this be done in a way that enhances deep cognitive work?

While the work environments of knowledge workers vary widely across different industries, this paper focuses on the office workspace and to the knowledge worker operating a computer system on a desk to complete a range of work activities. The primary artifacts in this context are the computer, and the top surface of the desk as shown in Figure 1. The knowledge worker may be a novice or an expert who may be succeeding on tasks, making errors, overloaded or distracted. Tasks may vary from writing documents, responding to email, computing calculations or debugging software code. Secondary artifacts in the office space such as lamps, toys, books, posters and windows may be used for changing mood, taking a break or sparking creativity.

The organization includes all supervisors and team-mates who interact with the knowledge worker to perform work tasks using the artifacts. Situations may be normal, abnormal or emergency scenarios that determine the relevant interactions during operation time. Situation complexity is often caused by interruptions, context-switching, and high workload conditions.

One predominant problem in environments such as these is the propensity for the knowledge worker becoming distracted to the detriment of work completion. Due to the reduced attention or impaired alertness, the inevitable net result is increased human error, and reduced ability to work safely, and productively [3] and decreased job satisfaction [4]. Human-centered design of the humanoid robot as a collaborative team-mate may enhance the task performance and psychological well-being of the knowledge worker. The robot should not cause stress and discomfort but incorporate social signals, cues and norms into the planning and control architecture to ensure psychological safety of the knowledge worker [5].

In this paper, we present the first steps of research into developing a computational model for a humanoid robot to support the cognitive needs of knowledge workers by increasing the duration and depth of their cognitive work. The second section of this paper outlines the state-of-the-art research on the psychological state of flow, neurofeedback-driven nudges, and human-robot interaction. In the third section, we present a design rationale for the smart personalized humanoid assistant. A demonstration of the agent-based framework was conducted to analyze the effects of the neurofeedback processing, and action planning modules. In the final section of the paper, we discuss the implications of neurofeedback-driven nudges, and conclude with limitations of our research, and suggestions for future work.

2. Related Work

2.1 Deep Work and Flow State

Knowledge workers are exposed to both internal and external distractions and interruptions that lead to suboptimal productivity, increased stress and dissatisfaction with their work. The average knowledge worker spends 30 percent of their daily time on email, and another 30 percent on team communication and Internet activities [2]. A significant portion of lost time is spent in context-switching, where additional time is lost before an interrupting task is started, and after it ends [6].

Distractions between episodes of knowledge work are a significant source of lost productivity and dissatisfaction due to the difficulty in accomplishing work goals. If these periods of distraction can be decreased or eliminated, the resulting ability to focus and complete tasks may positively impact on productivity.

Flow is defined as "the state in which people are so involved in an activity that nothing else seems to matter; the experience itself is so enjoyable that people will do it even at great cost, for the sheer sake of doing it" [7]. In the workplace, flow occurs when individuals, teams or organizations operate with optimal focus, and perform without apparent effort, which yields a heightened sense of motivation, intrinsic satisfaction, and peak outcomes [8, 9].

Flow experiences are not always easy to attain and sustain, but developing the ability to control attention may be an effective way to find and maintain flow [10]. Concentration is a significant component of achieving flow [11]. To establish the flow state, the human must focus attention on the activity, and goals at hand. Our hypothesis is that if the operator is aware of their state of concentration, they will be better able to drive towards deeper levels.

2.2 Neurofeedback and Nudges

Attention focus is a trainable skill that may result in reaching the flow state [12]. One way to train sustained attention is to provide a sensitive feedback signal so human users can learn to sense upcoming lapses earlier and prevent them from occurring in behavior [13]. Neurofeedback training has produced outcome gains in sustained attention [14], and has been used for the purpose of cognitive enhancement and as a therapeutic tool [15].

Neurofeedback is a category of biofeedback that is based on brain electrical signals recorded by bio-sensors placed on the head of humans. Neurofeedback involves the collection of the brain signals, classification and presentation of a stimulus to make individuals aware of their brain states and how to self-regulate certain brain-based processes [16]. In general, effective use of biofeedback requires specialized equipment to convert physiological signals into meaningful cues or a trained biofeedback practitioner [17].

To be effective working with the most difficult knowledge tasks in a domain, human workers often rely on external aids such as rituals, reminders and mental hacks, which this work collectively calls "nudges". This set of work-related nudges are used mostly on an ad-hoc basis to propel workers to deep levels of concentration and achievement. Nudges are used in behavior science to positively influence people while preserving their freedom of choice, and without engaging their deliberative capacities [18, 19].

The purpose of the reinforcement learning loop shown in Figure 2 is to help the knowledge worker achieve, and maintain a flow state while performing tasks. The aim is to have the Nao humanoid monitor and mirror the human's state so that it can effectively motivate them with nudges when they drift off task, and become invisible when their task performance is in the flow state.

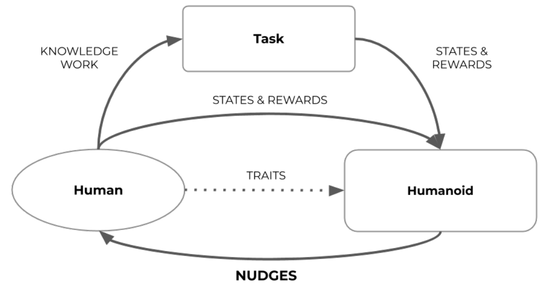


Figure 2. The reinforcement learning control model depicts the human completing work tasks in an operational environment that allows interactions with the humanoid agent. Percepts of human state and traits represent the agent's observations over time. The reward signal is obtained based on the humanoid's likeness to the human's subsequent flow state, and task completion. The humanoid performs nudge actions that consistently target stimulation towards deep flow.

The proposed nudges incorporate sound, physical motion, and other visual cues, and aims to combat boredom, fatigue, anxiety, distractions and habitual behaviors that may lead to under-performance during knowledge work. The nudges are delivered using traits of transparency and expressiveness versus autonomy and accuracy [20], good automation etiquette versus socially-agnostic behavior [21], and empathetic verbal apology over sterile error codes [22, 23].

Andujar et al [24] proposed that adapting humorous smart technologies in the workplace can be beneficial for employee retention and the improvement of employees' positive emotional state. The designers of humanoid interactions are recommended to consider how to detect negative responses resulting from the uncanny valley effect (where people react with unease seeing humanoid behavior that closely, but not perfectly, matches human behavior), and design robot behaviors to mitigate this by using humor or apology to overcome the robot's disagreeable appearance [25, 26, 27].

2.3 Cognitive and Collaborative Robots

Cognitive robotics involves the use of bio-inspired methods for the design of sensorimotor, cognitive, and social capabilities in autonomous robots [28]. Reggia et al [29] designed a cognitive humanoid robot framework with components that center on top-down control of a working memory that retains explanatory interpretations that the robot constructs during learning. The core function of our humanoid robot is to learn about the mental and emotional states of the knowledge worker and respond in a way to support their cognitive needs.

Collaborative robots are designed to physically interact with humans in a shared workspace instead of replacing them [30]. The humanoid robot proposed in this research provides personalized interacts with different knowledge workers while conducting work tasks in their workspaces. This interaction model is non-competitive co-operation by mutual understanding where authority is traded between human, and collaborative robot [31].

The sense of presence, and enjoyment that people feel with a robot can be manipulated by changing its social abilities [32]. In particular, the aesthetics, functionality, embodiment, situatedness, and morphology of humanoid robots can affect its interactions [33]. For example, Jo et al [34], found that interactions with physical and virtual humanoid robots produced a statistically significant effect in increasing human creativity. Automation inaccuracy can degrade human trust, comfort, and acceptance [35]. For these reasons, the humanoid's expressions and feedback are designed as a clear imitation of the knowledge worker's flow state.

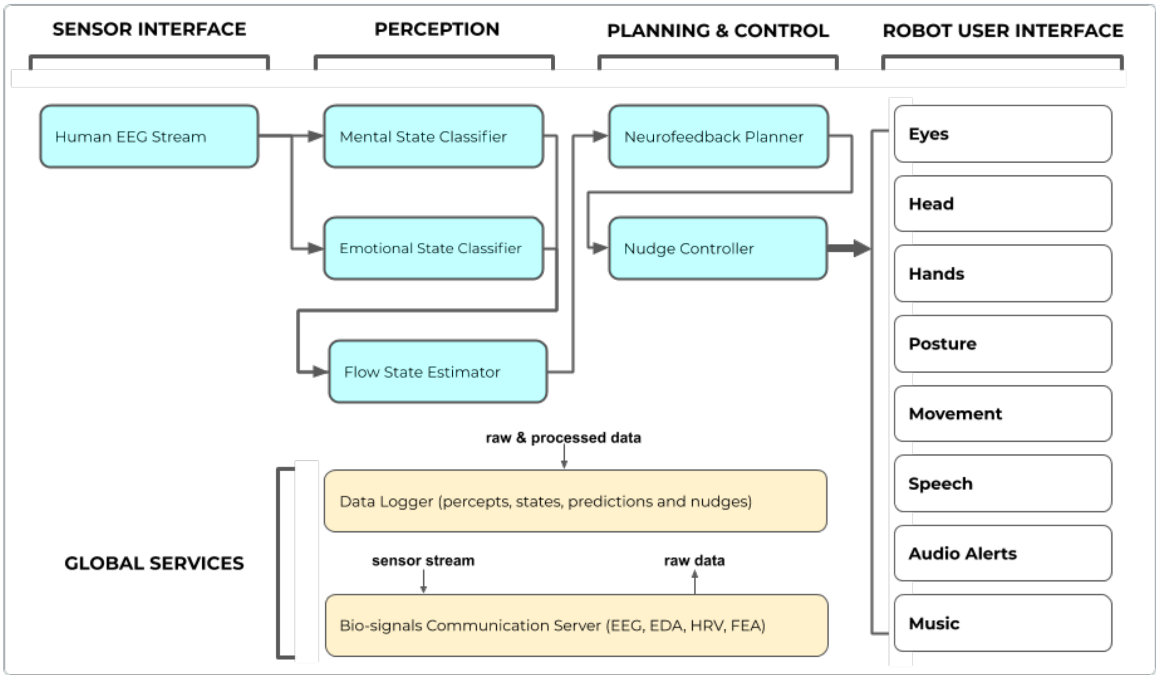


Figure 3. Pipeline architecture of the neurofeedback-driven humanoid where the modules process data on server and client threads in parallel.

3. Deep Work Humanoid Implementation

The humanoid obtains its bio-sensing capability from modules that process raw brain sensor data, classify human state, and plan appropriate nudges. The humanoid listens to the nudge controller, and executes nudges through expressive actions with its effectors.

3.1 Human EEG Stream

The human electroencephalogram (EEG) stream is collected from four EEG electrode bio-sensors on the Muse headband [36]. The sensor locations using the standard 10-20 coordinates are analogous to the prefrontal regions (AF7 and AF8) and the temporal regions (TP9 and TP10). The raw EEG data are sampled at 220 Hz on the four channels. The raw EEG data are transported to the server database via a mobile device interface that is capable of being synchronized with motion and other bio-signals.

3.2 Mental and Emotional State Classifiers

Noise and artifacts such as blinks and jaw clenches are filtered from the raw EEG data using a bandpass filter between 1 Hz to 75 Hz. The filtered EEG data are segmented into 4-second epochs without overlap. A short-term Fourier transform on each segment produces power spectral density (PSD) features in 5 frequency bands (delta, theta, alpha, beta and gamma). PSD features are classified using a recurrent neural network (RNN) with long short-term memory (LSTM) units. The mental states are engaged or disengaged. The emotional states are bored (B), happy (H) or anxious (A).

3.3 Flow State Estimator

In our research, the flow concept is operationalized with two dimensions: human skill and task challenge. Flow is modeled as the region where there is a balance between the human skill and task challenge. Flow is estimated as a discrete state (Distracted$_{\mathrm{B}}$, Shallow Flow$_{\mathrm{B}}$, Distracted$_{\mathrm{A}}$, Shallow Flow$_{\mathrm{A}}$, Shallow Flow$_{\mathrm{H}}$ and Deep Flow$_{\mathrm{H}}$) measured 4 seconds after presentation of the nudge.

3.4 Neurofeedback Planner

A common approximation to reinforcement learning (RL) problems is to ignore noise, which assumes full observability by learning and planning in observation space rather than a latent state space [37]. However, such approximations break down when using the Nao robot due to non-determinism in the actuators [38], and the EEG signals are stochastic due to sensors noise and artifacts. Therefore, the RL control model in Figure 2 is a discrete-time partially observable Markov decision process (POMDP). A Deep Q Network (DQN) is used to obtain the policy that maximizes the expected sum of rewards. The policy is used to plan and map the optimal nudge to the estimated flow state of the knowledge worker.

3.5 Nudge Controller

In order to help the knowledge worker maximize on deep work, the humanoid nudges with gestural feedback, reminders of task goals, and, in some cases, modifies the challenge level of the task relative to the skills of the knowledge worker. Nudges are conveyed in the form of expressive actions that the robot executes based on probabilistic computations of percepts, predictions and rewards. These humanoid expressions that are shown in Figure 4 represent stimuli to deliberately influence the knowledge worker by mimicking their mental and emotional state while motivating transition to deep flow.

3.6 Global Services

The architecture has a set of global services for logging percepts of raw data, PSD features, classified states, predictions and nudges. Streams of bio-sensor packets are transferred and aggregated via Bluetooth and User Datagram Protocol (UDP). The bio-sensor packets are synchronized and logged with discrete timestamps. Nudges are communicated from the server over Wi-Fi to the robot.

3.7 Robot User Interface

Only dependable robot architectures can be accepted for supporting "human-in-the-loop" conditions and human–robot teams [39]. Humanoids are a special case because they intrinsically present multiple control points for grasping, moving the head for perception, assuming postures, walking, and so on [40]. We hypothesize that a tangible interactive robot with the form factor of a warm, competent, and sensitive humanoid will be acceptable, and effective in knowledge work scenarios. The Nao humanoid supports the required human-like motor capability and has a high level of expressiveness that is capable of delivering a variety of nudges in Figure 4, which makes it suitable for our human-robot teamwork function. According best practices for social robots [26, 41, 42], we designed the humanoid to exhibit the following social characteristics:

- perceive, and express emotions;
- communicate with body language, and sound;
- exhibit a distinctive personality, and character;
- perceive natural cues (gaze, facial expression, head posture).

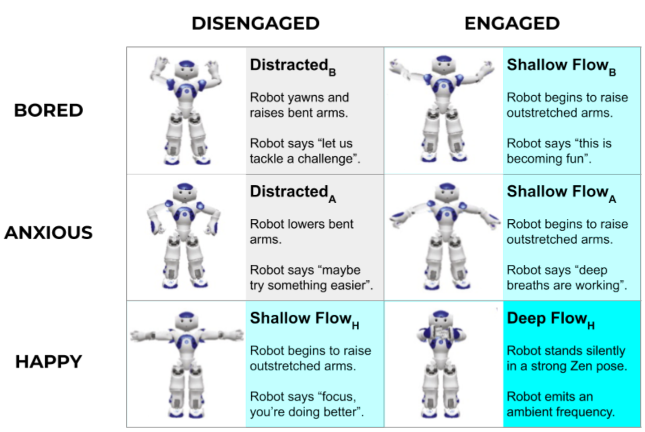


Figure 4. The six nudge categories of the Nao robot represent a sparse encoding that correlates expressions to the flow state, which is a product of mental and emotional state of the knowledge worker.

The expressions of the robot in Figure 4 are timely interactions with the knowledge worker without overt commands. These expressions strategically auto-encode the flow state of the knowledge worker into the metaphor of a Zen practitioner. The fluid movements of the humanoid are coupled with silence, music or motivational phrases spoken by the robot. Within the six categories of expressions, there is granularity of the phrases and sounds to improve salience and mitigate monotony. Silent motions avoid distracting the knowledge worker unnecessarily, and simple phrases minimize the cognitive load required to understand the robot.

According to Breazeal [41], the importance of feedback, and the readability of expression in this process cannot be underestimated in human-robot interaction. As the human applies the social model to understand the robot, they are constantly observing the robot's behavior, and manner of expression to infer its internal states. This allows the person to predict, and understand the robot's behavior only if the robot's expression is readable (the intended signal is appropriately interpreted by the human). The robot's expression reliably maps to the internal state being expressed, and this internal state adheres to the human's mental model of the robot.

4. Demonstration

In this section we review a demonstration of the architecture, showing how the pipeline architecture ingests classified mental and emotional states from EEG data, and then decides which nudge to produce on the humanoid robot. The synthetic dataset used in this demonstration simulated 2,000 trials i.e 100 work-hours. The trials were randomly sampled from 18 types of knowledge work scenarios. Each scenario type exhibited an overarching emotional state e.g. boring, anxious or happy. In order to simulate different frequencies of distractions, the scenarios were combined with the following types of perturbations in mental state:

- Engaged for 3 minutes
- Disengaged for 3 minutes
- Engaged for 90 seconds then disengaged for 90 seconds
- Disengaged for 90 seconds then engaged for 90 seconds
- Engaged then disengaged alternating every 1 minute
- Engaged then disengaged alternating every 12 seconds

The 3-minute scenario graphically depicted in Figure 5 shows a situation where a knowledge worker is in an overarching happy emotional state but experiencing intermittent 12-second distractions. Given this processed data from the humanoid's sensor and perception layer, the resultant flow state is handed to the neurofeedback planning and nudge control layers then an expression is communicated to the robot's user interface.

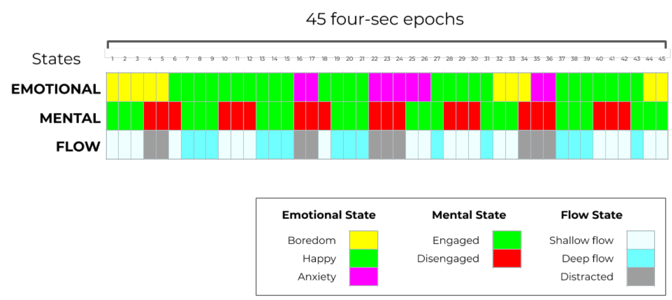


Figure 5. Visualization of a knowledge work scenario showing the emotional, mental and flow states over 45 epochs of sensor data after pre-processing and classification in the humanoid's software pipeline.

We adopted a Q-Learning approach as the baseline reinforcement learning (RL) control method for the neurofeedback planner. The RL method was implemented in Python to approximate the action-value function of the optimal policy. The Q-Learning algorithm used temporal differences to update the scores its Q-Table.

The data was split 50:50 into training and test sets, and reused across all experimental cases. The ε–greedy algorithm was used to make use of the exploration-exploitation tradeoff, and a linear function was used to decrement ε over the scenarios. Three reward functions were used to generate the different policies. The state-based function calculated reward based on the value of the resulting state if it was predicted. The distance-based reward function used the distance between the resulting state and the predicted state. The combo function represented a weighted combination of both measures. The nudges from the neurofeedback planner were tested on the Nao humanoid via the nudge controller. The following tables summarize the nudges resulting from the computational model for discussion.

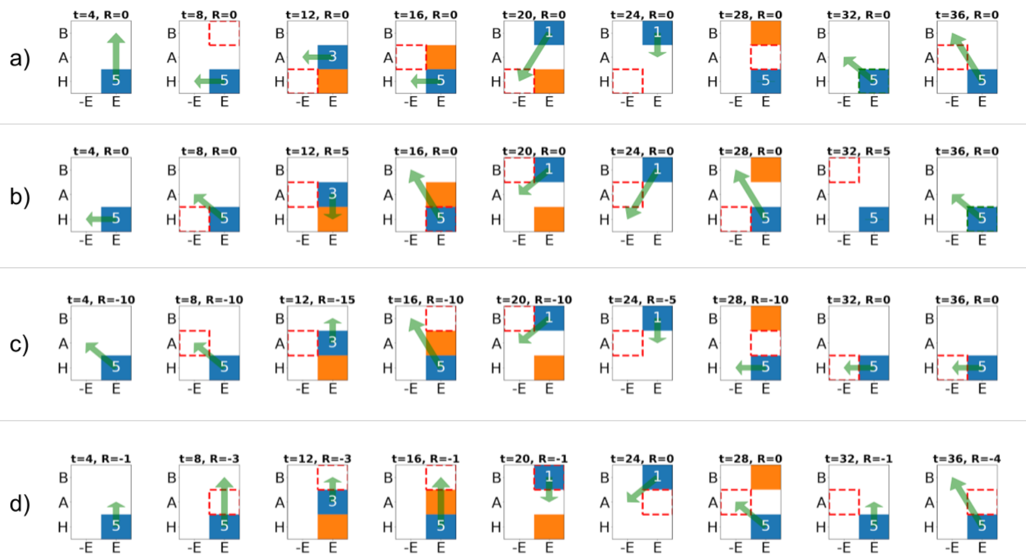


Figure 6. Comparison of policies for knowledge work scenarios across 9 four-sec epochs. The plots represent (a) before the humanoid was trained, and after training with (b) state-based reward, (c) distance-based reward, and (d) a combined state- and distance-based reward. The green arrows depict the attempted nudge. The blue cell represents the current observation, and the orange cell represents the previous observation. The dashed square highlights the previous nudge, where red indicates an incorrect prediction and green indicates a correct prediction.

4.1 Discussion

Figure 6 shows a comparison of three policies that were learned by the agent based on three different reward functions. Nudges in the baseline (a) appeared to be random and independent of observations. The state-based reward function in section (b) shows a different set of nudges, which yielded low rewards on the predictions. The distance-based reward function in section (c) depicted a more constrained set of nudges, which were closely related to the current observation. The combo-based reward function demonstrated that it is possible to combine benefits from multiple reward signals.

The nudges used in this demonstration were delivered every 4 seconds during the given knowledge scenario. When executed on the Nao, even though some nudges were silent, the number of nudges was relatively highly across the scenarios on average. This is likely to make the humanoid a distraction to maintaining flow and deep work. When the humanoid is deployed, an alternative approach to implement the neurofeedback planner would be to only nudge after (1) a stabilized period of the flow state, (2) after dwelling in distraction for 10 seconds, and (3) after task completion. The nudge after completing the scenario may be planned by classifying the knowledge worker's flow performance over the scenario and recommending the next task from a task list based on its estimated duration and challenge relative to the knowledge worker's skill.

5. Conclusion

This main goal of this research was to develop a computational model that effectively applies human-robot interaction in the deep knowledge work domain to augment human performance. The architecture presented in this paper offers a lightweight framework for investigating biofeedback-sensitive aids to deep cognitive work. Our hope is that this framework lowers the barrier for future research into the efficacy of different nudge regimes.

Breazeal, Aly, and others are adamant that the long-term emotional effects of humanoid interactions on humans need to be investigated [43, 44]. Issues arising of ethical concern include overreliance on automation [45], and potential manipulation of people through social robots [46]. Some AI algorithms may suffer from inaccuracies in the training data or may possess vulnerabilities to errors, and biased rules, which may lead to inaccurate profiling [47].

Notwithstanding the challenges and ethical issues, there is a possible design of the future where robots do not replace humans but enhance them to make a positive difference in the world. The impact of effective neurofeedback-driven humanoids that support deep work is potentially significant in terms of economic productivity, and social well-being of knowledge workers.

5.1 Future Work

Real human- and real humanoid-in-the-loop experiments are an exciting part of the next steps. Beyond the theoretical experiments reviewed in this paper, humanoids with simulation-trained neurofeedback planners will be compared with those trained with real world experience. Then we can determine if the humanoid's effectiveness may be improved by tailoring personalized nudges that fit the traits and preferences of specific knowledge workers.

Future studies will examine “attention” in greater detail, e.g. nudges that help knowledge workers to train their focus on task relevant stimuli, and boost their immunity to distraction. It is noted that a human can be focused and attentive on a distracting stimulus. This differentiation will require laboratory research with motion tracking to provide an objective measure of how attention is directed, and controlled by the knowledge worker during tasks.

Advanced work in feature engineering can improve the humanoid's accuracy and responsiveness. For example, the environment's reward signal that is currently controlled by task list completion and humanoid flow similarity may be augmented by human-centered metrics such as joy and positive affect, which may increase the reliability of the neurofeedback planner.

On the hardware side, the sensor interface may be extended to include different bio-signals such as facial expression, heart-beats, skin conductance, etc. The purpose of multimodal bio-signal fusion would be to increase classification accuracy, and improve the explainability of our deep learning models.

References

[1] J. Zumbrun, "The rise of knowledge workers is accelerating despite the threat of automation," *The Wall Street Journal*, May 4, 2016.

[2] C. Newport, *Deep Work: Rules For Focused Success in a Distracted World*. Hachette Uk, 2016.

[3] W. G. Sirois, "The myths & realities of fatigue: Reducing the costs, risks, and liabilities of fatigue in 24-hour operations," *Circadian White Paper*, 2009.

[4] A. Gazzaley and L. Rosen, *The Distracted Mind: Ancient Brains in a High-tech World*. Cambridge, Ma: Mit Press, 2016.

[5] J. Hu, B. Erdogan, K. Jiang, T. N. Bauer, and S. Liu, "Leader humility and team creativity: The role of team information sharing, psychological safety, and power distance.," *Journal of Applied Psychology*, vol. 103, no. 3, p. 313, 2018.

[6] S. Monsell, "Task switching," *Trends in Cognitive Sciences*, vol. 7, no. 3, pp. 134–140, 2003.

[7] M. Csikszentmihalyi, "Flow: The psychology of optimal performance," 1990.

[8] M. Csikszentmihalyi, "Play and intrinsic rewards.," *Journal of Humanistic Psychology*, 1975.

[9] J. Nakamura and M. Csikszentmihalyi, "The concept of flow," in *Flow and The Foundations of Positive Psychology*, pp. 239–263, Springer, 2014.

[10] M. Csikszentmihalyi, "Flow, the psychology of optimal experience, steps towards enhancing the quality of life.," 1991.

[11] M. Koufaris, "Applying the technology acceptance model and flow theory to online consumer behavior," *Information Systems Research*, vol. 13, no. 2, pp. 205–223, 2002.

[12] Y.-y. Tang and M. I. Posner, "Attention training and attention state training," *Trends in Cognitive Sciences*, vol. 13, no. 5, pp. 222–227, 2009.

[13] T. D. Megan, J. D. Cohen, R. F. Lee, K. A. Norman, and N. B. Turk-browne, "Closed-loop training of attention with real-time brain imaging," *Nature Neuroscience*, vol. 18, no. 3, p. 470, 2015.

[14] J. H. Gruzelier, "Eeg-neurofeedback for optimising performance. i: A review of cognitive and affective outcome in healthy participants," *Neuroscience & Biobehavioral Reviews*, vol. 44, pp. 124–141, 2014.

[15] S. Enriquez-geppert, R. J. Huster, and C. S. Herrmann, "Eeg-neurofeedback as a tool to modulate cognition and behavior: A review tutorial," *Frontiers in Human Neuroscience*, vol. 11, p. 51, 2017.

[16] J. S. Kreutzer, J. Deluca, and B. Caplan, *Encyclopedia of Clinical Neuropsychology*. Springerlink, 2018.

[17] D. L. Frank, L. Khorshid, J. F. Kiffer, C. S. Moravec, and M. G. Mckee, "Biofeedback in medicine: Who, when, why and how?," *Mental Health in Family Medicine*, vol. 7, no. 2, p. 85, 2010.

[18] R. H. Thaler and C. R. Sunstein, *Nudge: Improving Decisions About Health, Wealth, and Happiness*. Penguin, 2009.

[19] Y. Saghai, "Salvaging the concept of nudge," *Journal of Medical Ethics*, vol. 39, no. 8, pp. 487–493, 2013.

[20] M. Johnson, J. M. Bradshaw, P. Feltovich, C. Jonker, B. Van Riemsdijk, and M. Sierhuis, "Autonomy and interdependence in human-agent-robot teams," *IEEE Intelligent Systems*, vol. 27, no. 2, pp. 43–51, 2012.

[21] C. A. Miller, "Trust in adaptive automation: The role of etiquette in tuning trust via analogic and affective methods," in *Proceedings of The 1st International Conference on Augmented Cognition*, pp. 22–27, Citeseer, 2005.

[22] C. Ray, F. Mondada, and R. Siegwart, "What do people expect from robots?," in *Proceedings of The IEEE/Rsj 2008 International Conference on Intelligent Robots and Systems*, pp. 3816–3821, IEEE Press, 2008.

[23] Y. Iwamura, M. Shiomi, T. Kanda, H. Ishiguro, and N. Hagita, "Do elderly people prefer a conversational humanoid as a shopping assistant partner in supermarkets?," in *Proceedings of The 6th International Conference on Human-robot Interaction*, pp. 449–456, ACM, 2011.

[24] M. Andujar, A. Nijholt, and J. E. Gilbert, "Designing a humorous workplace: Improving and retaining employee's happiness," in *Advances in Affective and Pleasurable Design*, pp. 683–693, Springer, 2017.

[25] B. R. Duffy, "Anthropomorphism and the social robot," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 177–190, 2003.

[26] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 143–166, 2003.

[27] L. Huang, D. Mcdonald, and D. Gillan, "Exploration of human reactions to a humanoid robot in public stem education," in *Proceedings of The Human Factors and Ergonomics Society Annual Meeting*, vol. 61, pp. 1262–1266, Sage Publications Sage Ca: Los Angeles, Ca, 2017.

[28] N. M. Seel, *Encyclopedia of The Sciences of Learning*. Springer Science & Business Media, 2011.

[29] J. A. Reggia, G. E. Katz, and G. P. Davis, "Humanoid cognitive robots that learn by imitating: Implications for consciousness studies," *Frontiers in Robotics and Ai*, vol. 5, p. 1, 2018.

[30] J. E. Colgate, M. A. Peshkin, and W. Wannasuphoprasit, "Cobots: Robots for collaboration with human operators," in *Proceedings of The Asme Dynamic Systems and Control Division*, p. 433, American Society of Mechanical Engineers, 1996.

[31] G. Boy, *Orchestrating Human-centered Design*. Springer Science & Business Media, 2012.

[32] M. Heerink, B. KrÖSe, V. Evers, and B. Wielinga, "The influence of social presence on acceptance of a companion robot by older people," *Journal of Physical Agents*, 2008.

[33] B. Miller and D. Feil-seifer, "Embodiment, situatedness, and morphology for humanoid robots interacting with people," *Humanoid Robotics: A Reference*, pp. 1–23, 2016.

[34] D. Jo, J.-g. Lee, and K. C. Lee, "Empirical analysis of changes in human creativity in people who work with humanoid robots and their avatars," in *International Conference on Learning and Collaboration Technologies*, pp. 273–281, Springer, 2014.

[35] A. Hamacher, N. Bianchi-berthouze, A. G. Pipe, and K. Eder, "Believing in bert: Using expressive communication to enhance trust and counteract operational error in physical human-robot interaction," in *Robot and Human Interactive Communication (Ro-man), 2016 25th IEEE International Symposium On*, pp. 493–500, IEEE, 2016.

[36] P. Bashivan, I. Rish, and S. Heisig, "Mental state recognition via wearable eeg," 2016.

[37] M. L. Littman, "Reinforcement learning improves behaviour from evaluative feedback," *Nature*, vol. 521, no. 7553, pp. 445–451, 2015.

[38] A. Hasanain, T. Weekes, M. Person, K. Paul, Y. Chang, A. Rothman, Z. Rui, R. Rahil, M. Syed, C. Wodd, and Others, "Models and framework for supporting humanoid robot planning & exploration," 2018.

[39] A. De Santis, B. Siciliano, A. De Luca, and A. Bicchi, "An atlas of physical human–robot interaction," *Mechanism and Machine Theory*, vol. 43, no. 3, pp. 253–270, 2008.
[40] A. De Santis, "Modelling and control for human-robot interaction: Physical and cognitive aspects," in *2008 IEEE International Conference on Robotics and Automation*, Citeseer, 2008.
[41] C. Breazeal, "Toward sociable robots," *Robotics and Autonomous Systems*, vol. 42, no. 3-4, pp. 167–175, 2003.
[42] K. Dautenhahn, T. Fong, and I. Nourbakhsh, "A survey of socially interactive robots," 2014.
[43] A. Aly, S. Griffiths, and F. Stramandinoli, "Metrics and benchmarks in human-robot interaction: Recent advances in cognitive robotics," *Cognitive Systems Research*, vol. 43, pp. 313–323, 2017.
[44] C. Breazeal, "Emotion and sociable humanoid robots," *International Journal of Human-computer Studies*, vol. 59, no. 1-2, pp. 119–155, 2003.
[45] R. Parasuraman and V. Riley, "Humans and automation: Use, misuse, disuse, abuse," *Human Factors*, vol. 39, no. 2, pp. 230–253, 1997.
[46] R. Subramanian, "Emergent ai, social robots and the law: Security, privacy and policy issues," *Journal of International Technology and Information Management*, vol. 26, no. 3, pp. 81–105, 2017.
[47] T. Ring, "Privacy in peril: Is facial recognition going too far too fast?," *Biometric Technology Today*, vol. 2016, no. 7-8, pp. 7–11, 2016.