Original Research Article

Comfort-Aware Motion Planning for Robot-Assisted Feeding

Sidharth Maheshwari

Singapore American School, 40 Woodlands Street 41, Singapore

ABSTRACT

Robot-assisted feeding systems offer significant promise for individuals with partial motor impairment, yet delivering food comfortably to a human mouth remains a complex challenge. This work introduces a novel approach to motion planning in assistive feeding settings which both models and optimizes for human comfort. A comfort cost function was developed that integrates trajectory smoothness, velocity, and a novel "closeness" metric grounded in proxemics. This cost function serves as a guide to a new motion planning sampling strategy and a cost-aware shortcut smoothing for post-processing. Through simulation experiments in PyBullet across various realistic assistive feeding scenarios, the proposed methods generated paths with 17% decrease in comfort cost compared to standard planners making it more suited for adapting to human comfort. While the approach shows substantial gains in environments with a wider free configuration space, improvements were limited in environments where the free configuration space was narrower. These findings suggest a promising foundation for comfort-aware motion planning, with implications for improving autonomy and independence in robotic caregiving.

Keywords: Assistive Robotics; Motion Planning; Human-Robot Interaction; Sampling-Based Planning; Human Comfort Modeling; Robot-Assisted Feeding; Proxemics

INTRODUCTION

For the over 13 million people with upper body limitations, facing difficulty with "using their fingers to do things such as pick up a glass or grasp a pencil" (1), and the over 1 million people who "could not perform this at all" (1), robot-assisted feeding presents a promising solution to both maintaining independence and continuing daily tasks such as feeding. Transferring food from a held spoon to another human's mouth is a deceptively complicated task, typically performed by human caregivers. Notably, the trajectory taken needs to account for the personal space and comfort of the human while being fed. While it seems simple for humans, modeling and finding paths comfortable for humans prove to be a challenging problem. If algorithms can be found which find comfortable paths, it allows for robot-assisted feeding to be a viable alternative to relying on caregivers, regaining autonomy and independence for the portion of the population which struggle with upper-body motor skills tasks.

Current solutions, like the Obi (2), are commercially available for robot-assisted feeding. However, the Obi relies on kinesthetic learning, where a caregiver or external adult demonstrates a desired feeding motion manually by guiding its spoon (2). While this does have advantages, kinesthetic learning notably lacks the

Corresponding author: Sidharth Maheshwari, E-mail: sidharth.a.mahesh@gmail.com.

Accepted October 28, 2025
https://doi.org/10.70251/HYJR2348.36118131

versatility and adaptability of sampling based-planning.

The proposed new method both models and develops a novel algorithm, building upon standard sampling based-planning techniques, which both quantifies and optimizes for human comfort in assistive feeding settings. Since it relies on sampling based-planning, the proposed algorithm has no dependence on demonstrations like the current commercially available solutions. A new metric for closeness to a human was developed, which models the idea of personal space for the human while a spoon approaches their mouth. Then, this novel closeness metric can be combined with metrics for speed and smoothness to create a weighted cost function which models the comfort of a trajectory. Then, using the novel planning method coupled with and without a modified smoothing algorithm and testing it against regular sampling-based planning algorithms allows us to measure the proposed algorithm's efficacy in producing comfortable paths.

It was found that the proposed closeness function worked well to model the comfort of a path visually. For the novel algorithm, it was found that it was effective in producing more comfortable paths than a standard BiRRT in scenarios where the free configuration space is wider. Yet, when the free configuration space was narrower, both our novel algorithm and smoothing did not produce paths significantly more comfortable than their standard counterparts.

LITERATURE REVIEW

This paper has three primary areas of related work: Proxemics-based human-robot interactions, assistive robotic feeding systems, and sampling-based motion planning.

Proxemics-based human-robot interactions

Proxemics, the study of spatial relationships in social interactions, is a crucial factor to consider when designing socially acceptable robotic systems. *Edward T. Hall*, who coined the term, modeled differing zones of personal space, providing a framework that can inform robot behavior in shared human environments (3). Several studies have explored the modeling of proxemic zones for social robotics applications (4). *Daza et al.* developed a proxemic-aware system that adapts robot behavior based on distance and orientation metrics (5). *Camara et al.* found proxemic zones, with a focus on movement, are non-circular and depend on agent heading and dynamics (6). The research develops upon this idea, modelling human comfort through varying zones, adjusting for severe differences in approach and closeness.

Robotic Systems for Assistive feeding

Belkhale et al. conducts a similar study, both modeling and developing an algorithm which optimizes for comfort and efficiency (7). However, this covers the generalized process of scooping and picking up the food, rather than focusing on the transference of the food itself. *Rhodes et al.* focuses on finding comfortable trajectories, but instead proposes a technique for optimising kinesthetic learning algorithms (8). *Gordon et al.* focuses on providing common insights on implementing these assistive feeding algorithms as a proper, viable system (9). Of these three, this paper derives from *Belkhale et al.* (7) the most, but it differs in the novel way it models proximity and the area of focus for the feeding algorithm. While *Belkhale et al.* (7) focused on balancing comfort and efficiency during food acquisition and transfer, this work focuses post food acquisition on food delivery, explicitly modeling human comfort during that stage. Unlike *Rhodes et al.* (8), which improved kinesthetic learning through human demonstrations, the approach followed is sampling-based which can operate without demonstration data, rather relying on a cost-optimized BiRRT that directly optimizes for user comfort.

Sampling-based motion planning

Sampling-based motion planning has become a standard approach for solving high-dimensional configuration space problems in situations where explicit modeling is unfeasible. The Rapidly-exploring Random Trees (RRT) algorithm constructs a tree from a start configuration by sampling random points and incrementally connecting them to the nearest node on the tree (10). This allows rapid exploration of unexplored areas of space, guaranteeing probabilistic completeness often at the cost of yielding jagged or suboptimal trajectories. A widely used extension, Bidirectional RRT (BiRRT), improves convergence by simultaneously growing trees from both the start and end configurations and attempting to connect them (10). Notably, both RRT and BiRRT inherently treat all paths as equal, without regard to efficiency, path equality, or human comfort. This limitation has motivated cost-aware extensions, where planners incorporate domain-specific metrics to bias path generation. This paper builds directly on that foundation by extending BiRRT with a comfort-based cost function and cost-aware smoothing, providing motion planning algorithm and shortcut smoothing primitives usable in any context you can model a cost function for.

METHODS AND MATERIALS

In this section, the methods used to investigate path planning algorithms which optimize assistive feeding are detailed. The problem is approached in following steps:

Firstly, mathematically model the idea of human comfort with a cost function, specifically taking into account smoothness, speed, and closeness. While modeling of smoothness and speed are commonly known, a novel approach to representing closeness to the human, modeling the idea of personal space, is introduced.

Secondly, a new path planning algorithm that optimizes for the aforementioned cost function is introduced. This algorithm will build upon previous sampling-based path planning algorithms, namely Bi-Directional RRT (10) to account and optimize for a cost function using a set of precomputed "good" configurations.

Finally, a novel path optimization algorithm which builds upon shortcut smoothing (10) to optimize for the aforementioned cost function is introduced. Together, it culminates into a proposal for an effective mathematical modeling of human comfort and two algorithms which can be used to optimize for any cost function.

Problem Statement

This paper will address the problem of finding trajectories that, from a given start and end configuration for a particular manipulator, find paths that optimize human comfort for food delivery. Key assumptions are that the food is already held in the robotic manipulators' end effector and the human remains stationary throughout the motion planning process.

Let m be the degrees of freedom for the robotic manipulator. Let j_i represent the joint value for the i^{th} joint in the manipulator, where $1 \leq i \leq m$. Let a joint configuration q represent an array of joint values such that the value q_i corresponds to the joint value for joint i, taking the form q = ($j_1, j_2, \ldots, j_m$), q ∈ R^m. Let Q_{free} be the set of q that are collision-free for the robot. A trajectory T is a set of joint values and times $T = \{(q_1, t_1), (q_2, t_2), \ldots, (q_k, t_k)\}$ where $q_i \in Q_{free}$, $t_i \in R$, and $t_n < t_{n+1}$.

Cost function is defined as $C(T) = \alpha S(T) + \beta V(T) + \gamma L(T)$, where $S(T)$, $V(T)$, $L(T)$, are functions which each represent metrics of smoothness, speed, and closeness respectively, and α, β, and γ, are adjustable parameters. The functions for smoothness and speed are commonly known. For smoothness $S(T)$, discrete sum of θ_i, the angle between consecutive configurations is used. For path speed $V(T)$, discrete path length in configuration space is used as a surrogate. How the closeness function, $L(T)$, is developed, is defined later in this section.

The problem at hand is to propose a motion planning algorithm which minimizes the above cost function, thus maximizing human comfort.

Closeness Metric

The function for the closeness of a trajectory, $L(T)$, was designed to utilize the idea of personal space from the human perspective. The position of the end effector with respect to the rest of the human's body is used to develop the closeness metric. Inside the human's personal space, the end effector should minimize the angle with the mouth such that it is visible and comfortable to the human at all times. Outside, it should try and minimize closeness to the rest of the body as a whole.

Firstly, what is included in the model of personal space needs to be defined. If the end effector is in a particular radius D from the mouth, it is important to include it as a part of personal space. Let the position of the end effector for a given configuration q be $x(q)$, p_m be the position of the mouth, and v_m be the vector normal to the mouth. Based on this, let $v_{me} = x(q) - p_m$. If $\|v_{me}\| < D$, the end effector is considered to be in the chosen personal space. Otherwise, if the end effector is far away from the mouth but close to the rest of the body, it is also a part of the defined personal space. If the component of v_{me} parallel to the mouth axis is found, it should serve as a measure for the proximity to the rest of the body. Let $proj_u v$ be the component of vector v parallel to u. If $\| proj_{v_m} v_{me} \| < R$, where R is a configurable parameter, then it should be included in the defined personal space.

If the end effector is in defined personal space, it should try its best to align itself with the mouth axis to maintain user comfort. Thus, one would want to penalize the angle between the vector from the mouth to the end effector, v_{me}, and the mouth axis itself. A small angle keeps the end effector in a visible and comfortable area, whereas a large angle typically indicates a violation of personal space. This angle is represented as the angle between vectors v_{me} and v_m. It can be computed with the following formula, $\cos^{(-1)}\left(\frac{\overrightarrow{v_m} \cdot \overrightarrow{v_{(me)}}}{|\overrightarrow{v_m}\|\overrightarrow{v_{(me)}}|}\right)$. Thus, the cost for a configuration found to be in the personal space, as seen above, should be $\cos^{(-1)}\left(\frac{\overrightarrow{v_m} \cdot \overrightarrow{v_{(me)}}}{|\overrightarrow{v_m}\|\overrightarrow{v_{(me)}}|}\right)$.

If the end effector is not in the personal space, it would be preferred to keep it away from the rest of the body as a whole. This can be done by finding the distance between our end effector and the human. Let the human be represented by a set of links H. The distance between our end effector's position, $x(q)$, and the human can be represented as $d = \min_{h \in H} \| x(q) - h \|$, where h is any link in H. Here it is preferred to penalize low distances more than high distances. So, the cost for a configuration q which is not in personal space can be set to e^{-Cd}, where C is a configurable parameter.

Taken together, our overall closeness metric becomes

$$\mathrm{L}(T) = \begin{cases} \cos^{-1}\left(\frac{\overrightarrow{v_m} \cdot \overrightarrow{v_{(me)}}}{|\overrightarrow{v_m}\|\overrightarrow{v_{(me)}}|}\right) & \text{if } \| v_{me} \| < D \text{ or } \| proj_{v_m} v_{me} \| < R \\ e^{-Cd} & \text{else} \end{cases}$$

Notably, the closeness metric relies on both the position of the end effector and the distance to the human. The position of the end effector is calculated through standard forward kinematics, using homogeneous linear transformations to model the effect of each joint value on the end effector to output a final position for the end effector given a configuration q.

Cost-Optimizing BiRRT

Let Q_{obs} be the set of configurations q which result in a collision for the manipulator. Path-planning algorithms look for trajectories in Q_{free} from a configuration q_i to a configuration q_f. Modeling Q_{obs} completely can be challenging, especially in nonstatic environments. Sampling-based planning avoids this problem by random sampling of possible configurations, using collision detection functions to determine whether the configuration is in Q_{free} or Q_{obs}, constructing a path from q_i to q_f through configurations it finds to be in Q_{free}. This sparsely models Q_{obs} without constructing it entirely, allowing adaptability and scalability for the path planning algorithm.

Within sampling-based planning algorithms, Rapidly-exploring Random Trees (RRT) are a popular choice. The RRT algorithm builds a tree of sampled nodes in Q_{free}, connecting each newly sampled node to the nearest node already existing in the tree. This characteristic allows RRTs to rapidly explore all parts of Q_{free} evenly. Since RRTs favor unexplored areas, they are effective in finding a path from any q_i to q_f.

Bidirectional RRT, a variant of the RRT algorithm, builds upon the RRT by extending a tree not only from the initial configuration q_i, but also from the final configuration q_f. Instead of extending to a random point, the second tree tries to extend to the newest node explored. This behavior decreases randomization and speeds up convergence for the algorithm, since trees extended from either end are far more likely to find a path. The standard bidirectional RRT algorithm looks like the following.

Algorithm 1: BiDirectional RRT: This algorithm grows two search trees, one from the start configuration and one from the goal, until they connect to form a feasible path.

1. Initialize two trees: T_i rooted at the start configuration q_i, and T_f rooted at the goal configuration q_f.
2. For a fixed number of iterations, randomly sample a configuration q_{target} from the configuration space.
3. Extend the active tree T_i toward q_{target} using the EXTEND procedure to generate a new node q_{new}.
4. If q_{new} reaches the opposite tree ($q_{reached}$), merge both trees and reconstruct the complete path from start to goal.
5. Continue iterating until either a valid path is found or the iteration limit is reached.
6. Return the resulting path if successful; otherwise, report failure.

For this BiDirectional RRT, typically the sampling function is simply a function that returns a random possible configuration q_{rand}. This, however, has varying costs for the cost function C, given that it simply searches for a single path from q_i to q_f, not accounting for costs whatsoever.

Our novel method proposes the following. The algorithm should optimize the cost function C. Firstly, precompute a set of "good" configurations based on the cost function. By biasing the algorithm towards these good configurations, the new algorithm is more likely to include good configurations in the desired final path. So, using a parameter for precomputed bias P, the sampling function can be biased to occasionally extend to these good configurations, as opposed to a purely random one.

This sampling results in paths which incorporate predefined "good" configurations, thus resulting in paths which tend to score lower on the cost function overall. This lower cost is associated with more comfortable paths, thus achieving our end goal of a path planning algorithm which optimizes human comfort.

Cost-Optimizing Shortcut Smoothing

These paths, while now accounting for cost, have no guarantee of being locally optimal. They may take long, convoluted routes through the configuration space, which may increase the cost and impact the comfort for the human user. To account for this, path-optimizing algorithms can be used.

Path-optimizing algorithms take in a path from a path planning algorithm and improve upon it. Shortcut smoothing, in particular, runs for a particular number of iterations, finding shortcuts in the path to see if a linear path from q_a to q_b is quicker than the path found from the path planning algorithm. Not only does this shorten paths but often smooths them too.

While this algorithm is effective in shortening the path, it does not account for our cost function. This means a shortcut could inadvertently create a path with a higher cost, resulting in a more uncomfortable movement. An algorithm is proposed which takes the same approach, but measures whether a smoothed algorithm has a higher cost before making the switch.

This modification results in a path P which is both shorter and results in a higher level of comfort, since the smoothing shortcut now takes into account the comfort metrics of the user. Note that since the comfort metric requires heavy computation in world space, the maximum iterations N should be altered accordingly.

Algorithm 2: Shortcut Smoothing (Cost Aware)

1. Randomly select two waypoints a and b such that $a < b$.
2. Attempt to connect $P[a]$ to $P[b]$ using a direct path through configuration space.
3. If any configuration collides, discard this shortcut.
4. Otherwise, form a new path P_{new} by replacing that segment.
5. If $C(P_{new})$ is lower than $C(P)$, update $P = P_{new}$.
6. Repeat for N iterations and return the resulting path.

RESULTS

Three different simulated environments were used, modeling possible assistive feeding scenarios, to both tune cost function to accurately model assistive feeding, and to test the quality of the paths our algorithms produced. The parameters of the cost function were refined until paths that are comfortable are reflected when compared to paths that are uncomfortable (eg. too jagged, invade personal space). This was done qualitatively by modifying the parameters till costs were relatively appropriate.

Once the cost function accurately modeled human comfort, two algorithms were tested to see whether they produced more comfortable paths. After a parameter sweep for precomputed bias P and a precomputed threshold T, it was found that the two variables were largely independent in relation to effect on path cost. Testing across all 3 scenarios revealed that the proposed modified BiRRT resulted in paths of lower cost, and thus higher levels of human comfort in scenarios where the free configuration space was not narrow. However, costs did not seem to differ significantly from a regular RRT when path planning with a narrower free configuration space. Comparison of the proposed modified smoothing method had lower costs across the board and similar final path lengths for all 3 scenarios when compared to regular shortcut smoothing.

Simulated Environment

The simulator used for setup of this experiment was PyBullet. Each scenario consisted of a mock assistive feeding scenario. A chair is pushed in such that the human is seated at a modeled table. Personal Robot 2 (PR2) is comfortably close such that it can reach both the start and end configurations through a variety of approaches. PyBullet provides functions for collision for the PR2 and closest point functions to find the minimal distance to the human for the cost function, both of which have their use cases detailed in the method. As seen in Figure 1A, Scenario 1 consists of PR2 directly on the right of the human, Figure 1B, Scenario 2 consists of PR2 directly across the table, and Figure 1C, Scenario 3 is the same setup as Scenario 1 but with obstacles mimicking a bowl and a glass in the way.

In all simulated scenarios, human subject is assumed to remain stationary during feeding. It is also assumed that the human can themselves sit in the chair without assistance, which implies that they can maneuver themselves, but their arms cannot or would preferably not be moved, for example when in sharp pain. The robot is sitting nearby in a place where both the initial configuration and the end configuration are reachable. To isolate comfort-related motion-planning effects, the

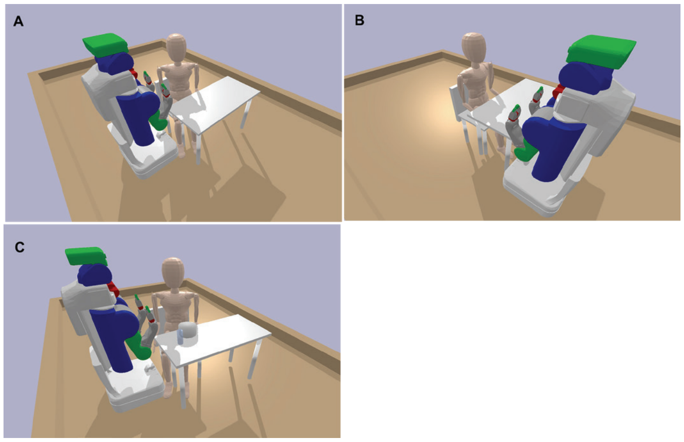

Figure 1. Simulated assistive-feeding Scenarios. (A) Scenario 1: PR2 directly to the right of the human. Start and end configurations are easily approachable with few obstacles in between. **(B)** Scenario 2: PR2 directly across the table. Narrower free configuration space, where the end configuration is harder to reach for the arm. **(C)** Scenario 3: PR2 directly to the right of the human but with obstacles mimicking a bowl and glass in the way.

initial state begins with the food already acquired and securely held by the robot's end effector.

Food to mouth stage is the most critical part of the assistive feeding process. Thus, creating comfortable paths from the food collected to the mouth is important. Under these conditions, a path from initial to final configurations which optimizes for human comfort would be useful for assistive feeding.

Notably, in order to test proposed planning algorithm thoroughly, each scenario also differs in the ease of planning from the start to end configuration. Scenario 1 represents the most basic scenario, where both the start and end configuration are easier to reach and there are few obstacles in between the two. This sets a baseline to tune the cost function, penalizing direct paths which come close to the human. Scenario 2 has a narrower free configuration space, where the end configuration is harder to reach for the arm. This allows us to test the efficacy of our modified planning algorithm in a scenario where the end path may be restricted. Finally, Scenario 3 introduces increased amount of obstacles between the start and end configurations, setup similar to a common dining setup, in order to test the efficacy of our modified algorithm with obstacles in the way.

These scenarios cover a range of possible situations where assistive feeding can be used. It covers different possible uses of a robot for assistive feeding, considering situations where paths are narrower or directly obvious. For each scenario, path planning on the right arm is run. The starting configuration is somewhere close to the table, post-scoop, and the end configuration is somewhere close to the human's mouth.

Cost Function Tuning

As previously established, our cost function is defined as weighted sum of closeness for every configuration, the smoothness, and the speed. This cost function takes the form $C(T) = \alpha S(T) + \beta V(T) + \gamma L(T)$, where $S(T)$, $V(T)$, $L(T)$, are functions which each represent metrics of smoothness, speed, and closeness , respectively, and α, β, and γ, are adjustable parameters. Notably, the closeness function $L(T)$ takes the following form

$$\mathrm{L}(T) = \begin{cases} \cos^{-1}\left(\dfrac{\overrightarrow{v_m} \cdot \overrightarrow{v_{(me)}}}{|\overrightarrow{v_m}|\,|\overrightarrow{v_{(me)}}|}\right) & \text{if } \| v_{me} \| < D \text{ or } \| proj_{v_m} v_{me} \| < R \\ e^{-Cd} & \text{else} \end{cases}$$

This results in a set of 6 parameters, 3 being weighting values of the cost function α, β, and γ, and 3 being modifiable values which shape the personal space D, R, and C. These 6 values were qualitatively adjusted till they reflected the idea of personal space well. As seen in Figure 2, this qualitative adjustment would be taking numerous paths and drawing them with their associated costs. In Figure 2, black lines represent lower cost paths, and red lines represent higher cost paths. Then, parameters are modified till the paths were accurate to an intrinsic sense of human comfort.

Based on the scale of numbers outputted by the functions, coupled with the fact that desired focus of

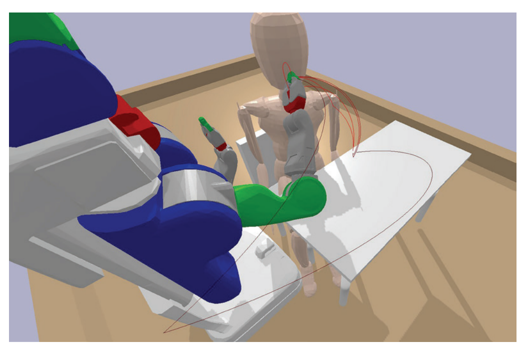

Figure 2. Qualitative Cost Test: Black lines represent lower cost paths, and red lines represent higher cost paths. Parameters modified till the paths were accurate to an intrinsic sense of human comfort.

algorithm was more on closeness and less on the other two functions, parameter values $\alpha = 1$, $\beta = 10$, and $\gamma = 2$ were selected. This resulted in a path that prioritized decreasing the closeness value the most, followed shortly by smoothness, and then lastly by the speed of the path.

Parameter Rationale and Sensitivity

The parameters used in the closeness metric were selected based on proxemics theory and validated through simulation. Guidance of *Hall's* intimate zone (3), which is 0 to 0.45m, provided the theoretical basis for determining personal space thresholds, D and R. While an initial baseline of $D = R = 0.2$ was tested, subsequent iterations showed that $D = R = 0.4$ offered a more accurate representation of the effective comfort boundary, producing trajectories that better aligned with visually "comfortable" paths.

The exponential penalty constant $C = 4.0$ was chosen through a qualitative sensitivity sweep, evaluating the effect of C on trajectory comfort and numerical stability. Lower values of C (< 3) under-penalized proximity intrusions near the human face, while higher values (> 5) led to steep gradients and unstable cost behavior. The chosen value provided a sharp yet stable penalty profile, ensuring that trajectories encroaching within the intimate zone were strongly discouraged.

With D and R and anchored in proxemics theory and C arrived through qualitative sensitivity testing, parameter choices were both theoretically grounded and empirically validated against qualitatively known "comfortable" configurations.

Precomputed Parameter Tuning

The modified BiRRT algorithm relies on two parameters, namely the threshold T for which a path is constituted as good in the precomputed set, and the precomputed bias P for which a precomputed algorithm is favored. Parameter sweeps of thresholds (represented in degrees with the mouth axis, since any configurations outside of personal space are ignored) and parameter bias was conducted. As seen in Figure 3A, the higher the precomputed bias, the higher the average cost. Furthermore, a precomputed bias in the range 0.01 to 0.96 was run. A finer tuned parameter sweep found 0.06 to work best. Similarly, as seen in Figure 3B, a threshold in the 10-degree range seemed to fit results the best.

Modified BiRRT Comparison

Testing modified BiRRT against BiRRT and RRT yielded the results listed in Figure 4 and Table 1. With 500 trials per algorithm and a precomputed set, findings suggest that the modified BiRRT standing alone, performs similar to a regular BiRRT. However, visualizing the paths reveals that the high cost modified BiRRT paths come from approaching the good configurations from a bad angle. So, while they would include these good configurations, in the process of getting to them they would end up decreasing comfort and increasing closeness. Using the Anytime variant, which allots a time limit to find the lowest cost path, overcomes this problem by finding multiple approaches to incorporate good configurations and selecting the one

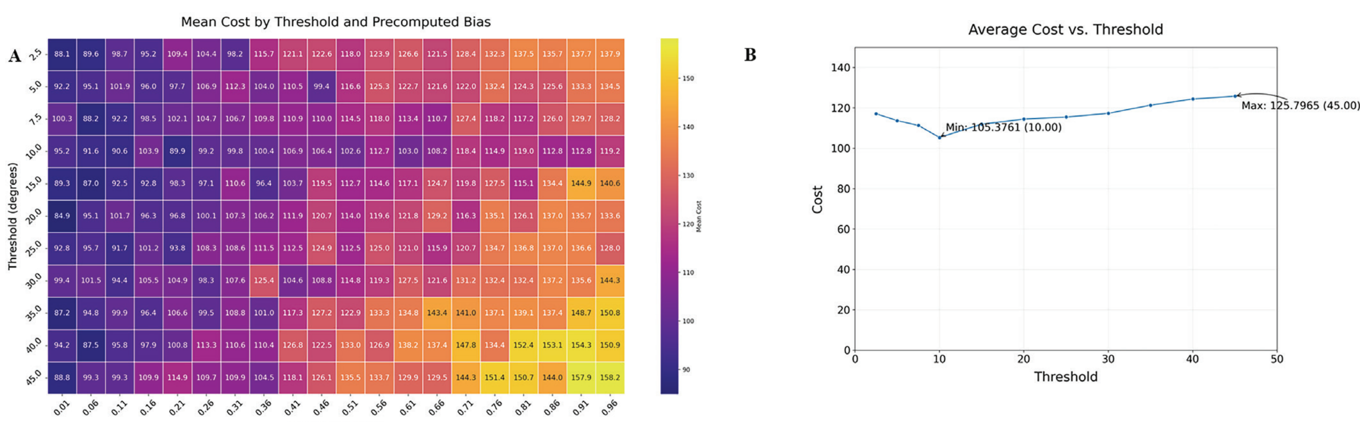

Figure 3. Precomputed Parameter Tuning – Parameter Sweeps. (A) Mean Cost by Threshold and Precomputed Bias: Parameter sweeps of thresholds (represented in degrees with the mouth axis) and parameter bias. Higher the precomputed bias, the higher the average cost. **(B)** Avg Cost vs Threshold.

that minimizes the closeness and maximizes comfort.

For Anytime variant, on average, it was observed there was 20.230% decrease in cost from RRT and a 9.982% decrease in cost from a regular BiRRT for Scenario 1 (Figure 4A, Table 1A). In a narrower free configuration space, where the target end configuration has fewer possible paths to approach due to the reach of the robotic arm, modified setup does not produce any notably more comfortable paths (Figure 4B, Table 1B). Finally, in scenarios with obstacles such as in Scenario 3, a similar improvement of 15.321% cost against RRT and 16.928% cost against BiRRT is seen (Figure 4C, Table 1C).

To evaluate whether these improvements were statistically significant, a one-way ANOVA was performed on comfort cost values across algorithms. Across all scenarios, the ANOVA indicated a significant effect of algorithm type on comfort cost ($p < 0.01$) suggesting that the differences in performances were unlikely due to random chance. Post-hoc Tukey HSD tests further showed that, in Scenarios 1 and 3, the Modified Anytime BiRRT achieved significantly lower costs than both the standard RRT ($p < 0.001$) and BiRRT ($p < 0.02$). In Scenario 2, no statistically significant differences were found among the algorithms ($p > 0.3$). These results statistically confirm the observed improvement by the Modified Anytime BiRRT in terms of human comfort for planned trajectories in Scenario 1 and 3.

Modified Shortcut Smoothing Comparison

Visualizing the better paths produced from the anytime variant found that a lot of paths considered would take an optimal path backward but would overshoot how far they would travel. This resulted in unnecessarily long

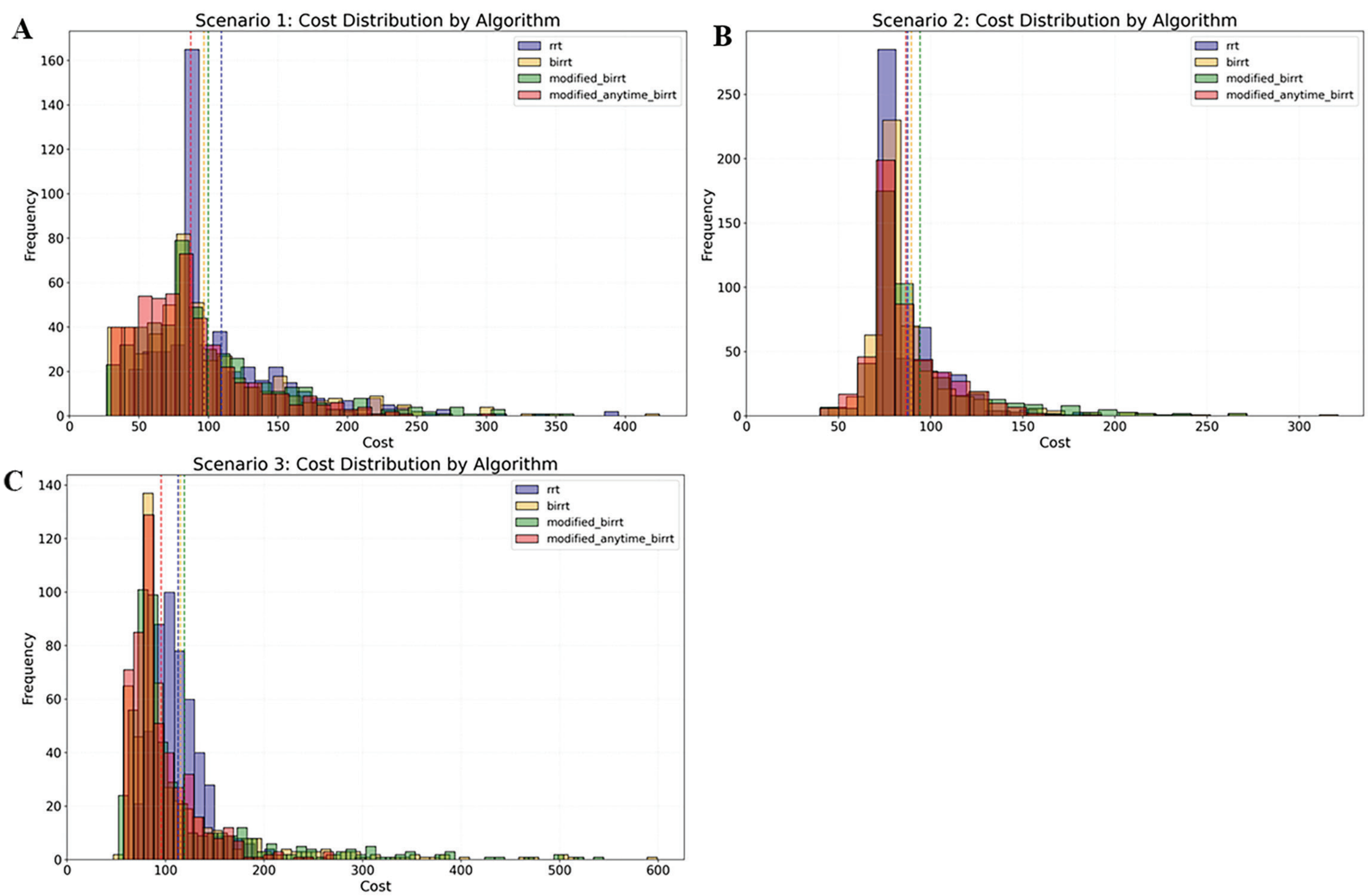

Figure 4. Modified BiRRT Comparison – Cost Distributions by Algorithm. (A) Scenario 1: PR2 directly to the right of the human. Start and end configurations are easily approachable with few obstacles in between. **(B)** Scenario 2: PR2 directly across the table. Narrower free configuration space, where the end configuration is harder to reach for the arm. **(C)** Scenario 3: PR2 directly to the right of the human but with obstacles mimicking a bowl and glass in the way.

Table 1. Modified BiRRT Comparison Metrics

Metric	RRT	BiRRT	Modified BiRRT	Modified Anytime BiRRT
A				
Average Cost	109.311	96.867	99.892	87.197
Cost St. Dev	49.265	55.338	56.462	41.605
Average Time Taken	1.854	1.344	1.177	9.611
Time Taken St. Dev	3.774	0.763	0.615	0.432
Average Nodes Sampled	204.884	91.344	94.226	151.178
B				
Average Cost	87.46	89.565	94.233	86.635
Cost St. Dev	19.663	30.562	33.583	20.744
Average Time Taken	1.459	0.903	1.076	9.582
Time Taken St. Dev	1.175	0.471	0.548	0.462
Average Nodes Sampled	118.396	74.01	81.846	104.03
C				
Average Cost	112.827	115.011	119.039	95.542
Cost St. Dev	25.985	74.774	78.118	33.611
Average Time Taken	1.105	0.806	0.877	9.508
Time Taken St. Dev	2.78	0.71	0.757	0.472
Average Nodes Sampled	123.478	53.758	58.43	99.102

(A) Scenario 1: PR2 directly to the right of the human. Start and end configurations are easily approachable with few obstacles in between. **(B)** Scenario 2: PR2 directly across the table. Narrower free configuration space, where the end configuration is harder to reach for the arm. **(C)** Scenario 3: PR2 directly to the right of the human but with obstacles mimicking a bowl and glass in the way.

path with little to no changes in comfort level. Shortcut smoothing addresses this problem by finding linear shortcuts in the number of configurations a trajectory goes through. Proposed modified version makes these long backward turns a lot more efficient. When running modified algorithms with cost aware shortcut smoothing algorithm and comparing it to standard RRT or BiRRT and standard shortcut smoothing, both forms of shortcut smoothing improve cost as compared to the unsmoothed versions. In Scenarios 1 (Figure 5A, Table 2A) and 3 (Figure 5C, Table 2C), modified shortcut smoothing coupled with proposed algorithms provide much more comfortable paths as a whole. In Scenario 2 (Figure 5B, Table 2B), where the configuration space was narrower, it is found that it is only slightly more effective than the typical shortcut smoothing algorithm, but both smoothing algorithms greatly improve the comfort of the paths, regardless.

Across all three scenarios, the modified BiRRT and its anytime variant perform strongly in reducing path complexity while retaining, or even improving comfort, for assistive feeding paths. While raw path-length reductions for Scenarios 1 and 2 are lower than those achieved by BiRRT, both modified methods achieve these paths at substantially reduced smoothing costs, indicating that they are able to generate solutions which balance efficiency and comfort. This tradeoff is important in assistive feeding setting, where abrupt motions caused by overly aggressive path shortening can be uncomfortable for the end user. Moreover, in Scenario 3, a scenario consisting of realistic obstacles in an assistive feeding setting, the strength of the modified approaches is most clearly evident: both versions not only achieve the highest path-length reductions

(44.8% and 42.1%, surpassing BiRRT's 40.0% and RRT's 33.9%) (Table 2C), but also maintain the lowest smoothing costs, with the anytime variant attaining the shortest final smoothed path length (19.93) (Table 2C). Notably, the magnitude of cost reduction was greater in Scenarios 1 and 3, where the free configuration space was wider than that of Scenario 2.

To evaluate whether these improvements were statistically significant, a one-way ANOVA was performed on smoothing cost and path-length reductions across algorithms. For all scenarios, the ANOVAs revealed a significant effect of algorithm type for smoothing cost ($p < 0.001$) and path-length reductions ($p < 0.001$), indicating that the differences in performances were unlikely to be due to random chance. For all scenarios, post-hoc Tukey HSD tests revealed that the Modified Anytime BiRRT and Modified BiRRT produced significantly lower smoothing costs than the regular RRT ($p < 0.01$) and the BiRRT ($p < 0.03$) algorithms. Similarly, post-hoc Tukey HSD tests revealed that Modified Anytime BiRRT and Modified BiRRT had higher path-length reductions overall ($p < 0.05$). These results statistically confirm the observed improvement by the proposed modified shortcut smoothing algorithm seen with Modified BiRRT and Modified Anytime BiRRT.

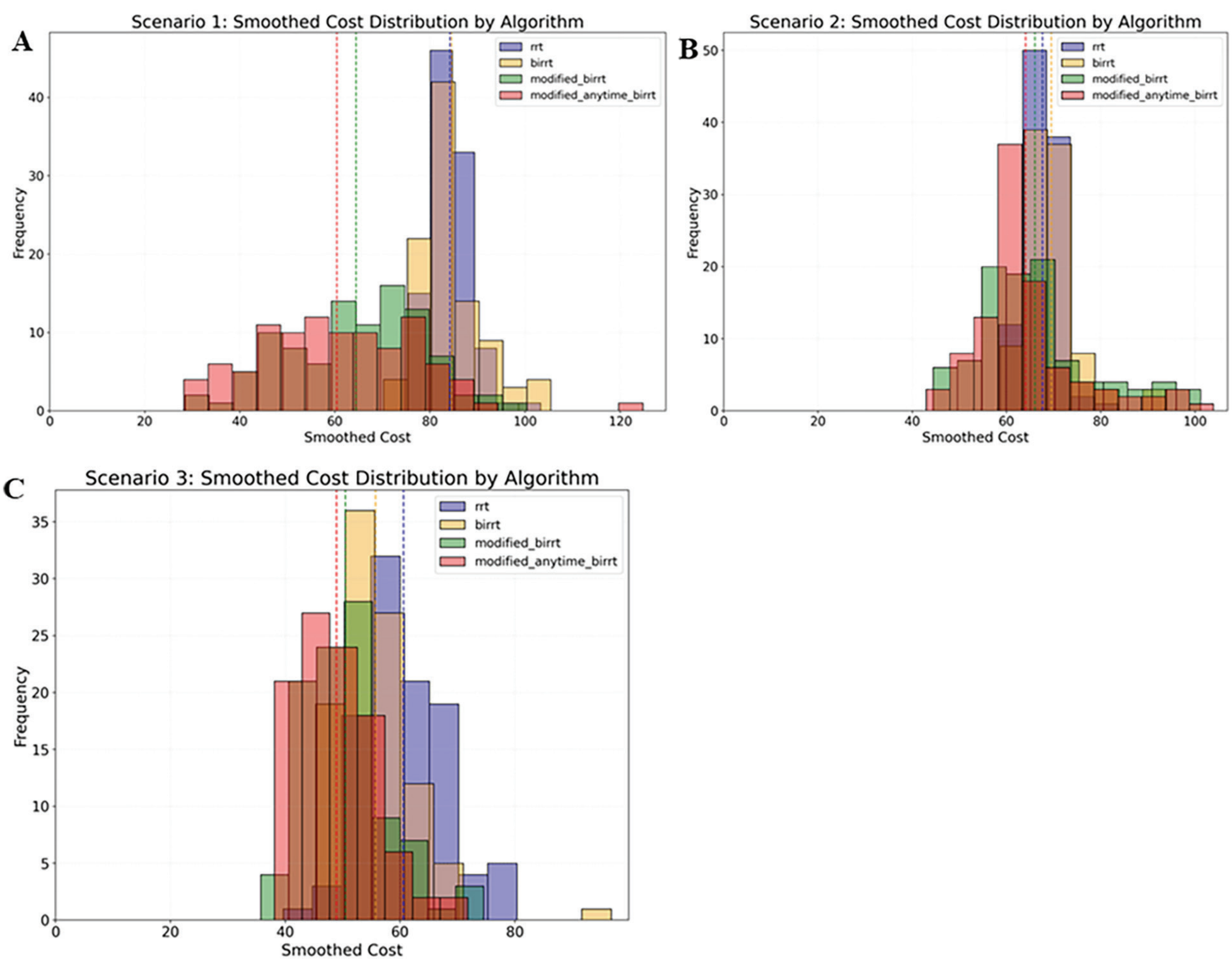

Figure 5. Modified Shortcut Smoothing Comparison – Smoothed Cost Distributions by Algorithm. (A) Scenario 1: PR2 directly to the right of the human. Start and end configurations are easily approachable with few obstacles in between. **(B)** Scenario 2: PR2 directly across the table. Narrower free configuration space, where the end configuration is harder to reach for the arm. **(C)** Scenario 3: PR2 directly to the right of the human but with obstacles mimicking a bowl and glass in the way.

Table 2. Modified Shortcut Smoothing Comparison Metrics

Metric	RRT	BiRRT	Modified BiRRT	Modified Anytime BiRRT
A				
Avg. Smoothing Cost	84.199	84.474	64.466	60.421
Smoothing Cost Std. Dev	4.320	6.713	14.481	16.669
Avg. Delta Path % Length	30.137%	44.645%	23.239%	21.004%
Delta Path Length % Std. Dev	19.948%	23.579%	22.105%	17.976%
Avg. Smoothed Path Length	38.894	38.112	63.01	65.52
B				
Avg. Smoothing Cost	67.608	69.532	66.021	64.041
Smoothing Cost Std. Dev	3.536	5.873	12.437	11.115
Avg. Delta Path % Length	17.183%	43.734%	27.662%	23.592%
Delta Path Length % Std. Dev	12.823%	23.674%	23.863%	18.848%
Avg. Smoothed Path Length	34.243	34.02	55.837	52.09
C				
Avg. Smoothing Cost	60.576	55.698	50.469	48.909
Smoothing Cost Std. Dev	7.417	6.796	7.402	6.739
Avg. Delta Path % Length	33.863%	39.994%	44.763%	42.068%
Delta Path Length % Std. Dev	10.173%	22.413%	21.808%	17.324%
Avg. Smoothed Path Length	22.777	22.44	21.32	19.93

(A) Scenario 1: PR2 directly to the right of the human. Start and end configurations are easily approachable with few obstacles in between. **(B)** Scenario 2: PR2 directly across the table. Narrower free configuration space, where the end configuration is harder to reach for the arm as a whole. **(C)** Scenario 3: PR2 directly to the right of the human but with obstacles mimicking a bowl and glass in the way.

DISCUSSION

Implications

System design, placement and practical implications: The three scenarios studied map naturally to common feeding contexts: Scenario 1 approximates bedside or side-seated assistance where the manipulator can approach along a broad lateral arc; Scenario 2 approximates across-table feeding with a narrow approach corridor; Scenario 3 reflects everyday dining clutter (bowl, cup) that constrains approach cones. The findings indicate that the physical placement of the robotic system relative to the user substantially influences the comfort of planned trajectories. In scenarios where the manipulator operated in wider free configuration spaces, the cost-aware planner produced consistently smoother and more acceptable motions. This suggests that system setup, including the adjustment of seating, table configuration, and robot base position, is a critical design factor. Careful placement reduces planning complexity and enables the trajectory optimization process to more effectively realize improvements in user comfort.

When the approach corridor to mouth is narrow (Scenario 2), the comfort-aware planner shows negligible improvements. This can be attributed to two factors. First, sampling diversity collapses: both standard and modified BiRRTs tend to discover similar, tightly constrained paths, leaving little room for the precomputed low-cost configurations to change the overall route. Second, lower effective manipulability near the end pose restricts local steering options and diminishes the impact of shortcut smoothing. In this environment, the closeness term often saturates, meaning the approach cone cannot be widened without collision or reach violation, so comfort-biased sampling provides smaller marginal benefit compared to wider configuration scenarios (Scenarios 1 and 3), where multiple lateral approach possibilities exist. These results highlight the importance of preserving approach diversity and goal-region redundancy during system setup.

A key assumption made is that the human stays stationary throughout the feeding process, but in real life scenarios there could be two major variations. Firstly, the human may shift posture. In such a scenario the algorithm can be executed in an online loop at a fixed rate during which the current head/mouth pose is estimated, the comfort cost is re-evaluated, and motion adjustment is triggered if the mouth position deviates beyond a threshold. The cost-aware shortcut smoothing could then be applied with a limited window so that the updates remain smooth and do not introduce abrupt motions. Secondly, because comfort is person-specific, the closeness parameters assumed may not be comfortable for all the users. In this case, the closeness parameters (*D, R, C*) and the cost function weights (α, β, γ) can be calibrated for each user during an initial training phase capturing person specific comfort preference with respect to approach angles and stand-off distances. After the parameters are set, they can be used as priors and adapted over time from passive observations.

Above considerations should be used for a deployable assistive-feeding behavior in real life scenario. Broadly, the analysis across scenarios suggests that comfort-aware planning is most effective when combined with thoughtful environmental configuration, including balancing geometry, reachability, and real-time adaptability, to translate algorithmic comfort gains into tangible user benefit.

Cost-aware planning as a primitive: The planning modifications introduced in this work, namely the sampling bias toward precomputed low-cost configurations and the incorporation of cost-awareness in smoothing, are generalizable across tasks. These methods provide a reusable framework whenever a task-specific cost function can be defined, such as in contexts where safety, efficiency, or social acceptability must be optimized. In assistive feeding, these primitives demonstrate the feasibility of embedding human-centered metrics directly into the planning process, with potential applicability to other domains of human-robot interaction.

Limitations

The absence of human-subject testing is the primary limitation of the present work and the next major step. The findings are based on simulations, which may not correlate with real life findings. Human comfort is an intrinsically defined concept, varying from human to human. So, to verify whether our mathematical model correctly accounts for personal space, it needs to be compared against multiple human accounts for whether it is personally comfortable or not. Further, each human may have differing comfort thresholds, requiring personalization for optimal modelling. Personalization of comfort parameters has not been accounted for in the paper. Real life user evaluations of comfort function and the algorithm are required to validate the thesis derived from simulations.

Another key limitation with the process was of the assumption that the human remains stationary throughout the feeding process. In a realistic assistive feeding interaction, users may shift posture, lean forward or backward, or react to the food being fed by the robot. Such dynamic reactions require consistent re-planning based on the human's pose. This is out of the scope of this paper.

Another limitation is that the closeness metric relies on the minimum distance to the human. While in simulation, finding this distance doesn't prove to be a challenge, in practice, even approximating it poses a problem. Furthermore, closeness metric also requires the position of the end effector in world space, a task which can be computationally heavy. Both of these variables, together, substantially increase computation time and reduce its applicability in a real assistive feeding scenario.

CONCLUSION

The research explored a novel path-planning approach that maximizes human comfort for robot-assisted feeding. Drawing from traditional sampling-based planners (BiRRT and RRT), the work makes following contributions. Firstly, it introduces a human comfort cost function which, in addition to smoothness and speed, incorporates a novel closeness metric, which is modeled on the idea of personal space. Secondly, it creates a novel cost-aware planning algorithm, modified BiRRT, which biases sampling towards precomputed low-cost configurations. Lastly, it creates comfort cost-aware shortcut smoothing algorithm that preserves comfort while simplifying trajectories. Experiments in three distinct PyBullet scenarios offer proof that the cost function can not only model human comfort, but cost-aware path planning algorithms can generate paths that are smoother, more considerate of personal space, and more comfortable for users.

The results across three simulated scenarios indicate when and how comfort gains materialize. Improvements are largest in wider free configurations (e.g. bedside/side-seated or cluttered but spacious dining contexts), and negligible in narrow configurations (e.g. across-table or narrow corridors). This implies actionable setup guidance. Caregiver seating, table layout, and robot base placement should preserve a wider free configuration space around the mouth so that the comfort-aware planner and smoothing can realize their full benefit.

Beyond feeding, the cost-centric formulation can be generalized towards assistive tasks that require optimizing social acceptability and safety margins near the face (e.g., grooming and utensil handovers).

The framework also outlines a practical path to deployment: integrate online re-planning under live mouth position estimates to accommodate user movement and personalize comfort parameters via brief calibration and adaptation over time. With these steps, after real life human validation, the proposed comfort-aware planning algorithm can serve as a robust primitive for assistive robotics in everyday settings.

Future Work

A major next step involves human-subject evaluation. While simulation results demonstrate the feasibility of comfort-aware planning, real-world testing with participants is essential to verify whether the modeled comfort cost aligns with perceived comfort and user acceptance. A pilot study comparing standard BiRRT trajectories with the proposed cost-aware planner could measure both subjective comfort ratings and objective metrics such as approach smoothness, clearance, and response to head movement. These experiments would provide critical data for refining the comfort model and validating its practical benefit in assistive-feeding settings.

Further research could be done to incorporate real life scenario where humans are not stationary during the feeding process. Future implementations could integrate real-time head or mouth pose tracking with short-horizon re-planning, allowing the comfort-aware algorithm to continuously adjust to human motion while maintaining smooth and natural trajectories.

Because human comfort may vary human by human, with each having different thresholds and preferences, further studies can be conducted to incorporate personalization for human comfort. The parameters of the closeness metric (D, R, C) and the comfort weights (α, β, γ) can be calibrated to each user through an initial setup and refined overtime, leading to personalized comfort-aware feeding.

An alternative approach, where humans can provide feedback, can be used to determine possible ways to model the cost of each path. On similar lines, human comfort modeling can be extended through the idea of Kinesthetic Learning. A kinesthetic approach to model and account for human comfort may yield better results in modeling something as complicated as comfort.

Finally, since our cost-optimization algorithm is generalized for any cost function, it can be used and utilized to solve any problem which is modellable with a cost function. Future work could explore cost-aware performance of the proposed algorithm across a variety of different applications, including but not limited to further human-robot interaction, precision control of robotic manipulators, and task-priority control. Further, exploring adaptive or learning-based cost functions could make the proposed framework even more versatile and adaptable in real-world environments.

ACKNOWLEDGEMENTS

I would like to thank my mentor, Prof Dmitry Berenson, whose guidance was fundamental to the success of this project.

FUNDING SOURCES

There was no funding used for this research.

CONFLICT OF INTERESTS

The author declares that there are no conflict of interests related to this work.

REFERENCES

1. Taylor DM. Americans with disabilities: 2014. Current Population Reports. *Washington (DC): U.S. Census Bureau.* 2018; P70-152. Available from: https://www.census.gov/library/publications/2018/demo/p70-152.html (accessed on 2025-04-11).
2. Obi Robot. Meet Obi: the Adaptive Eating Device – Eat Independently! [Internet]. Jacksonville (FL): Obi Robot; c2025. Available from: https://meetobi.com/ (accessed on 2025-04-11).

3. Hall ET. The Hidden Dimension. Illustrated reprint. New York: Knopf Doubleday Publishing Group; 1990. Available from: https://books.google.com/books/about/The_Hidden_Dimension.html?id=zGYPwLj2dCoC (accessed on 2025-04-11).
4. Takayama L, Pantofaru C. Influences on proxemic behaviors in human-robot interaction. In: 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems; 2009 Oct 11-15; St. Louis, MO. Piscataway (NJ): IEEE; 2009; p.5495-5502. doi: 10.1109/IROS.2009.5354145.
5. Daza M, Barrios-Aranibar D, Diaz-Amado J, Cardinale Y, Vilasboas J. An approach of social navigation based on proxemics for crowded environments of humans and robots. *Micromachines.* 2021; 12 (2): 193. doi:10.3390/mi12020193.
6. Camara F, Fox C. A kinematic model generates non-circular human proxemics zones. *Adv Robot.* 2023; 37 (24): 1566-1575. doi:10.1080/01691864.2023.2263062.
7. Belkhale S, Gordon EK, Chen Y, Srinivasa SS, et al. Balancing efficiency and comfort in robot-assisted bite transfer. arXiv:2111.11401 [cs.RO]. 2021. https://doi.org/10.1109/ICRA46639.2022.9812332
8. Rhodes T, Veloso M. Robot-driven trajectory improvement for feeding tasks. In: 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2018 Oct 1–5; Madrid, Spain. Piscataway (NJ): IEEE; 2018; p.2991–2996. doi:10.1109/IROS.2018.8593525.
9. Gordon EK, Jenamani RK, Nanavati A, Liu Z, et al. An adaptable, safe, and portable robot-assisted feeding system. arXiv:2403.04134 [cs.RO]. 2024. https://doi.org/10.1145/3610978.3641085
10. LaValle SM. Planning algorithms. Cambridge (UK): Cambridge University Press; 2006. Available from: http://planning.cs.uiuc.edu/ (accessed on 2025-04-11). https://doi.org/10.1017/CBO9780511546877