Tuning Cost Functions for Social Navigation

David V. Lu[1], Daniel B. Allan[2], and William D. Smart[3]

[1] Washington University in St. Louis, St. Louis, MO 63130, USA,
davidlu@wustl.edu,
[2] Johns Hopkins University, Baltimore, MD 21218, USA,
[3] Oregon State University, Corvallis, OR 97331, USA

Abstract. Human-Robot Interaction literature frequently uses Gaussian distributions within navigation costmaps to model proxemic constraints around humans. While it has proven to be effective in several cases, this approach is often hard to tune to get the desired behavior, often because of unforeseen interactions between different elements in the costmap. There is, as far as we are aware, no general strategy in the literature for how to predictably use this approach.

In this paper, we describe how the parameters for the soft constraints can affect the robot's planned paths, and what constraints on the parameters can be introduced in order to achieve certain behaviors. In particular, we show the complex interactions between the Gaussian's parameters and elements of the path planning algorithms, and how undesirable behavior can result from configurations exceeding certain ratios. There properties are explored using mathematical models of the paths and two sets of tests: the first using simulated costmaps, and the second using live data in conjunction with the ROS Navigation algorithms.

1 Introduction

Navigation is one of the fundamental tasks in mobile robotics. For robots with reasonable dynamics and operating speeds in indoor environments, efficient collision-free navigation is considered a solved problem from a practical standpoint. However, when humans are introduced into the environment, we must treat them differently than static obstacles by respecting their social norms, thus causing navigation to become more difficult.

Most path-planners in use today use a discretized costmap, where values in the costmap cells correspond to the "badness" of the robot being in that position. The path-planner then generates a path from the start position to the end that has the minimum accumulated cost. This will cause the robot, in many cases, to closely approach obstacles. While this is fine for furniture, it is not socially acceptable with people.

Using the full range of values in the costmap is vital for representing many social constraints for navigation algorithms. They represent general preferences or guidelines rather than hard and fast rules. As Kirby et al. [6] observed, "Human social conventions are tendencies, rather than strict rules," This approach discourages a subset of paths without disallowing them outright.

In practice, creating the desired behavior around people is surprisingly difficult. While previous researchers have tuned their parameters to create working configurations, there exists no general guide for how to do this to effect a specific change in robot behavior. Furthermore, as we show below, the resulting behavior is not always intuitive from a consideration of the individual costmap elements.

2 Related Work

Methods for modeling social preferences in costmaps have existed for some time. Most uses of soft constraints in costmaps have been for representing a person's personal space. Dautenhahn et al. [4] constructed recommendations for planning motions where humans would be comfortable based on live HRI trials, taking proximity, visibility and hidden zones into consideration. These were formulated into a costmap system by Sisbot et al. [13], creating a Gaussian based "human aware motion planner." Kirby et al. [6] used an algorithm that in addition to minimizing path distance and avoiding obstacles, modeled proxemics and behaviors like passing on the right into the costmap, also using Gaussians. Work by Svenstrup et al. [14] created even more complicated models of personal space, integrating a mixture of four different constraints modeled as Gaussians. [15] expanded this work to maneuver among a field of multiple people while moving toward a goal. Soft constraints are occasionally used for other fields like autonomous vehicles. Ferguson and Likhachev [5] used large constant valued areas to favor driving on the right side of the road and to avoid curbs. Among the latest work in this field, Mainprice et al. [9] have expanded their original model to a three dimensional costmap in order to control positioning during hand off tasks, taking safety, visibility and the human's arm comfort into consideration. Scandolo and Fraichard [12] have also created a complex model that included proxemic, visibility and motion models as Gaussians, and "interaction areas" as constants. Lu and Smart [8] also have modified robot behavior using intermediate costmap values to improve the efficiency of human task completion.

Most of the obstacles added to the costmaps follow Gaussian distributions or constant values (with the minor exception of the representation of the intimate personal space in the work by Scandolo and Fraichard [12]). Little to no discussion is given about how the authors of the previous works found the parameters that worked best with their system.

The problem of designing costmaps that result in human-like behavior bears some similarity to inverse reinforcement learning (IRL) [11, 1, 2]. In IRL, the goal is to induce an immediate reward function based on example behaviors that will result in a similar final policy. In our setting, this would involve recording a number of human trajectories, determining a suitable set of features to serve as a state space (in a similar manner, perhaps, to [7]), then applying IRL algorithms to learn the functions that would create the local costmap based on these features. While an approach like this offers the promise of automatic costmap construction, it depends critically on the example trajectories and on a good feature set being selected.

3 Problem Statement

Each planned path depends on two separate components: the costmap and the planning algorithm. The costmap is represented by a two-dimensional grid, where each grid cell has a value $f(x, y)$. Values above some predefined threshold are designated as “lethal” values which will result in collisions. Standard algorithms such as Dijkstra’s and A* are typically used as the path planner. However, if the total cost of a path is defined as the cost of the cells the path traverses alone, then the resulting path will be a very long path that avoids any cost. To avoid this scenario, wavefront planners are often used, which add a constant value, P, to each cell traversed to create a gradient from start to finish[3].

Formally, we define $C(p)$ as the total cost of the path p (including both the costmap costs and the path planning costs). Finding the best path involves minimizing the cost over all possible paths.

$$\min_{\forall \text{path} p} C(p) = \min_{\forall \text{path} p} \sum_{(x,y)\in p} \Big[f(x, y) + P \Big] \tag{1}$$

This definition for path cost assumes that each step in the path moves to another grid cell exactly 1 unit away, implying each cell is connected to its four immediate neighbors. We use this assumption throughout this paper, although it is possible to generalize it, for example, to also include diagonal moves.

For purposes of this paper, let us further refine the problem to reduce the number of cases we must consider. First, without loss of generality, let us consider paths that go from $(-n, 0)$ to $(n, 0)$. We further assume that there are no lethal cells in our costmap, since path planning algorithms already do a fine job of avoiding these.

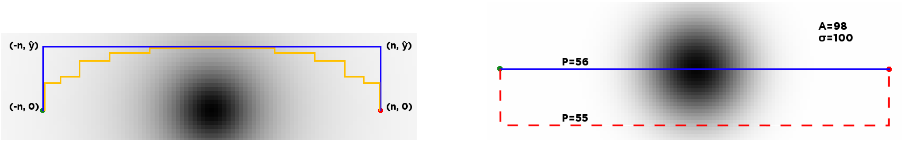
In addition to the actual planning problem, there is also the parameter tuning problem. We would like to be able to design robot behavior from a high level and not need to fiddle with the parameters endlessly to find the perfect balance for the desired behavior. One goal of this is to find functions in which the parameter space is either intuitive or limited to only the best possible values.

The interplay between the values in the costmap and the path planning constant turns out to be crucial for determining the course of the planned path. The duality of optimizing for path length or for path cost results in a continuum of different paths that could be considered optimal depending on the weighting of the two sides.

4 Mathematical Properties of Gaussian Obstacles

Our analysis focuses on the frequently-employed two-dimensional Gaussian distribution, defined as

$$f(x, y) = A \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \tag{2}$$



(a) The Bracket Shape

(b) Gaussian Cost Function

Fig. 1: Types of Paths - In 1a, an optimal bracket-shaped path in blue and a suboptimal path in yellow. In 1b, the discontinuity of optimal paths between $P = 55$ and $P = 56$.

The cost in each cell under the Gaussian depends on both the amplitude, A, and the variance, σ. One key feature of the Gaussian function (and all other monotonically decreasing functions) is that the optimal path is always bracket-shaped: from $(-n, 0)$ to $(-n, \hat{y})$ to $(n, \hat{y})$ to $(n, 0)$ for some $\hat{y}$. The proof of this is elided for space, but centers on the fact that all paths that reach $y = \hat{y}$ will have the same length (as seen in Figure 1a), and the bracket-shaped paths are the farthest paths of that length from the obstacle. Note that a direct path qualifies as a bracket with $\hat{y} = 0$.

The only variation in the paths is how far away from the obstacle they are, i.e. their $\hat{y}$ values. In this section we seek a relationship between the model parameters P, A and σ and the resulting distance of closest approach, $\hat{y}$. In order to cause the optimal path have to a smaller value of $\hat{y}$, the obvious strategies are to decrease the costs (decrease A or σ) or to increase the path constant P. This is generally true, but there exist certain conditions in which incrementally changing the parameters in this way will result in a drastically different path.

For instance, one would expect that increasing P incrementally would result in a gradual decline in $\hat{y}$. However, the resulting change sometimes is a discontinuous jump. Consider the paths in Figure 1b. With $A = 98$ and $\sigma = 100$, when P is increased from 55 to 56, one would expect $\hat{y}$ to undergo a small decrease. Instead, the path jumps suddenly through the origin ($\hat{y} = 0$). There is, in fact, no value of P which causes the optimal path to occur between these two paths.

4.1 Theory

Our goal is to show why there are certain parameters that result in one of three behaviors: **(1)** Minimum path cost is at $\hat{y} = 0$ **(2)** Minimum path cost is at finite $\hat{y} > 0$ **(3)** Minimum path cost is infinitely far away. The cost of a bracket path in terms of the model parameters and some choice of $\hat{y}$ is

$$\begin{aligned} C(p_{\hat{y}}) =& C(\text{segment } 1) + C(\text{segment } 2) + C(\text{segment } 3) \\ =& \Big[\hat{y}P + \sum_{i=0}^{\hat{y}-1} f(-n, i)\Big] + \Big[2nP + \sum_{x=-n}^{n} f(x, \hat{y})\Big] + \Big[\hat{y}P + \sum_{i=0}^{\hat{y}-1} f(n, i)\Big]. \end{aligned}$$

Now let us assume that we begin far from the obstacle, so $n \gg \sigma$. In this limit, the cost due to the obstacle on segments 1 and 3 is very small compared to the baseline cost and the cost along segment 2.

$$C(p_{\hat{y}}) \approx (2n + 2\hat{y})P + \sum_{x=-n}^{n} f(x, \hat{y}) \tag{3}$$

We are only concerned with the cost of each path in relation to other paths, so we will express the cost of some $p_{\hat{y}}$ relative to the cost of the direct path.

$$\begin{aligned} \mathit{\Delta} C(\hat{y}) =& C(\hat{y}) - C(0) \\ =& \Big[(2n + 2\hat{y})P + \sum_{x=-n}^{n} f(x, \hat{y})\Big] - \Big[(2n + 2(0))P + \sum_{x=-n}^{n} f(x, 0)\Big] \\ =& 2P\hat{y} + \sum_{x=-n}^{n} \Big[f(x, \hat{y}) - f(x, 0)\Big] \end{aligned}$$

When $\mathit{\Delta} C(\hat{y}) < 0$ for some $\hat{y}$, the direct path is not optimal.
With a Gaussian obstacle as f,

$$\mathit{\Delta} C(\hat{y}) = 2P\hat{y} + \sum_{x=-n}^{n} \left[A \exp\left(-\frac{x^2 + \hat{y}^2}{2\sigma^2}\right) - A \exp\left(-\frac{x^2}{2\sigma^2}\right)\right]$$

We have already assumed $n \gg \sigma$, the tails of the Gaussian will contribute negligibly, so we can approximate the sum over $x \in [-n, n]$ as the sum over all x, which has a simple solution. We use the following approximation.

$$\begin{aligned} \sum_{x=-n}^{n} A \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2}{2\sigma^2}\right) &\approx \sum_{x=-\infty}^{\infty} A \exp\left(-\frac{x^2}{2\sigma^2}\right) \exp\left(-\frac{y^2}{2\sigma^2}\right) \\ &= A\sigma\sqrt{2\pi} \exp\left(-\frac{y^2}{2\sigma^2}\right) \end{aligned}$$

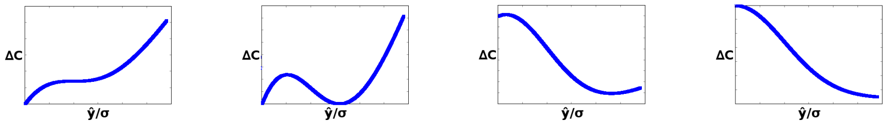
Finally, we have a closed-form expression for the cost of bracket path $p_{\hat{y}}$ compared to the direct path.

$$\mathit{\Delta} C(\hat{y}) = 2P\hat{y} + A\sigma\sqrt{2\pi}\left[\exp\left(-\frac{\hat{y}^2}{2\sigma^2}\right) - 1\right] \tag{4}$$

Now, we locate the $\hat{y}$ that minimizes $\mathit{\Delta} C(\hat{y})$.

$$\frac{d\mathit{\Delta} C}{d\hat{y}} = 2P - \frac{\hat{y}}{\sigma} A\sqrt{2\pi} \exp\left(-\frac{\hat{y}^2}{2\sigma^2}\right) = 0 \tag{5}$$

$$\frac{P}{A} = \sqrt{\frac{\pi}{2}} \frac{\hat{y}}{\sigma} \exp\left(-\frac{\hat{y}^2}{2\sigma^2}\right) \tag{6}$$



(a) $P/A = \sqrt{\pi/2e}$: The direct path $\hat{y} = 0$ minimizes $\varDelta C$.

(b) $P/A = 0.57$: A new global minimum is established at $\hat{y} > 0$.

(c) $P/A < 0.57$. $\hat{y}$ moves farther from the origin, and the minimum grows deeper.

(d) $P = 0$, and the global minimum is at $\hat{y} \to \infty$.

Fig. 2: Cost $\varDelta C$ relative to the cost of the direct path, as a function of ratio of the baseline cost-per-step P and the amplitude A of a Gaussian obstacle. The minimum of this function determines where the optimal path is.

The solution for $\hat{y}$ is related to the Lambert W-function [4], which cannot be written in closed form. Depending on the cost ratio P/A, it admits 0, 1 or 2 solutions. We can make several observations.

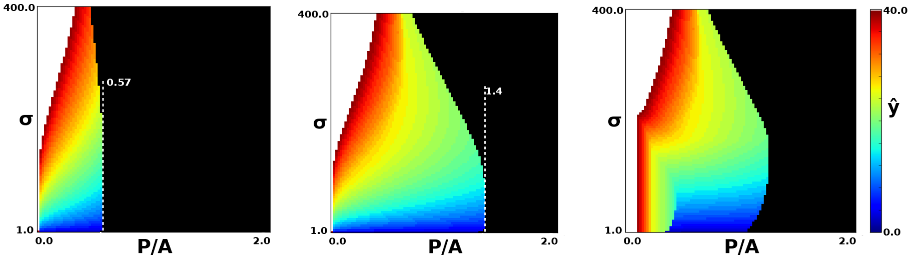
1. Eq. 6 peaks when $\hat{y}/\sigma = 1$, attaining $P/A = \sqrt{\pi/2e}$. If we set P and A such that $P/A > \sqrt{\pi/2e} \approx 0.760$, there is no solution to Eq. 6, and the only minimum of $\varDelta C$ occurs on the boundary at $\hat{y} = 0$ like in Figure 2a. The direct path is optimal.
2. For values of P/A less than .760, the inflection point at $\hat{y}/\sigma = 1$ decreases as well, creating a local minimum. This minimum remains a local minimum until $P/A \approx 0.57$ (Figure 2b), a point we determined numerically where the minimum becomes the global minimum. For values even less than 0.57 (Figure 2c), the direct path is no longer optimal and the center of the obstacle is avoided.
3. When $P = 0$, $\varDelta C(\hat{y})$ has a minimum at infinity (Figure 2d). This results in the path being as far away from the center of the obstacle as possible.

Thus, we have shown, with some simplifications, the mathematical underpinning for the relationship between the different parameters and why certain configurations lead to the three behaviors/values of $\hat{y}$ discussed at the beginning of this section.

5 Results from Simulation

To further explore the relationships between the parameters, we ran path planning algorithms over simulated costmaps as described in the Problem Statement section. Instead of showing all of the paths for each configuration, we represent the resultant $\hat{y}$ values in heatmaps as seen in Figure 3. As is evident from Equation 6, P and A are inversely related, a fact we could have surmised through

[4] See `http://en.wikipedia.org/wiki/Lambert_W_function`



(a) Four Connected (b) Eight Connected (c) Two Gaussians

Fig. 3: Heatmaps - The value of $\hat{y}$ as a function of P/A and σ. Black values represent paths where $\hat{y} = 0$; white values represent paths where $\hat{y}$ is infinitely far away. Redder values indicate higher values of $\hat{y}$ and bluer values represent lower values.

dimensional analysis. As long as the ratio $P{:}A$ remains constant, the value of $\hat{y}$ also remains constant (plots not shown). This means we can explore the entire parameter space as a two dimensional heat map relating that ratio to σ, which we did by varying the value of P.

The three distinct behaviors are seen in the different colorations. Configurations represented in black result in paths that move straight through the obstacle ($\hat{y} = 0$). On the opposite side of the spectrum, white configurations represent paths that are as far away as possible, i.e. the optimal $\hat{y}$ is infinitely large. The intermediate hues represent the finite values of $\hat{y} > 0$, with red values being the farthest away and blue values being the closest (smallest $\hat{y}$).

Let us begin the discussion with Figures 3a and 3b. Figure 3a was created using the von Neumann neighborhood (i.e. four connected) and corresponds with the math from the previous section. Figure 3b utilized the Moore neighborhood (i.e. eight connected), and thus the properties from the previous section need no necessarily apply. However, as evident in the two figures, both the von Neumann and Moore neighborhoods operate similarly, although with slightly different scaling. This lends substantiative support to our hypothesis that the properties operate similarly regardless of the precise path planning implementation used.

The next thing to note is the relationship between σ and $\hat{y}$ for a given P/A. For any given finite positive value of $\hat{y}$ we can decrease the variance to get a smaller $\hat{y}$. This leads us to our first general observation about the parameter space. **Decreasing the variance on a Gaussian will always lead to paths closer to the obstacle.** Similarly, we can assert that **Lowering the ratio $P : A$ will always increase $\hat{y}$ from a positive value to a greater value.**

However the inverse of these statements is not true. Increasing the variance will sometimes increase the distance from the obstacle, but can also lead to decreasing the distance down to $\hat{y} = 0$. **Increasing the variance on a Gaussian will only sometimes result in paths further from the obstacle.** Increasing

$P : A$ will always lead to to decreased $\hat{y}$ values, however, at some point, those values jump to $\hat{y} = 0$. This discontinuity means that **For a given σ, some values of $\hat{y}$ cannot be expressed.** If this were not the case, the right-hand edge of the colored areas in the heatmaps would be blue for the lowest $\hat{y}$. Similarly, some values of $\hat{y}$ cannot be expressed for a given ratio $P : A$. These two limitations lead to the most important observation about the parameter space: **Finding paths with the full range of $\hat{y}$ values cannot be achieved by tuning just one parameter.**

This point is reiterated by the fact that there are some values of $P : A$ that do not have any values of σ resulting in $\hat{y} > 0$. Our simulated experiments also validated that these dead zones occur when $P : A > 0.57$ when the grid is four-connected, which can be seen in Figure 3a and since these values result in no solution to Equation 6. Values of $P/A < 0.57$ generally result in non-zero $\hat{y}$, until the variance gets sufficiently large, which is where our initial assumption that $n >> \sigma$ breaks down, resulting in unpredictable behavior. Based on the results shown in Figure 3b, we have further estimated that for eight connected neighborhoods, the dead zone starts with $P/A > 1.4$.

6 Results using ROS Navigation

Much of this work is motivated by our experiences with the ROS navigation stack [10]. Initially, it did not have a way to directly input non-lethal obstacles, which made it difficult to model people's personal space. In a branched version of the software, we removed this limitation[8]. In the costmaps, the amplitudes can range from $[0, 254]$ with no limits on the variance. However, initially we were unaware that the default path planning constant was set so that $P = 50$, meaning that our options for A were limited, and we found it difficult to tune the parameters to get the passing distances we desired.

To further validate the principles in this paper, we used ROS navigation stack with real sensor data from Willow Garage's PR2. Paths were planned around a live human using a modified wavefront planner that performs interpolated gradient descent to determine a smooth path, i.e. not constrained to either the von Neumann or Moore neighborhoods. The results are shown in Figure 4 with three different values of A. As A increases, the value of $\hat{y}$ increases too, but then when A is very large, it contracts back down to $\hat{y} \approx 0$, as close to the person as possible. The full heatmaps are not included for space.

We found that even using the less restrictive planner, the same principles apply, although the constants have different values. Since costmaps and the discretized paths they produce are all approximations of the same continuous space, it is logical that the different grid connectivities would have similar results; however, it is nice to have the technical confirmation to back up the ideas. Not only is the solution more general, but it has the added benefit that the resulting smoother paths are more legible to people observing the robot.

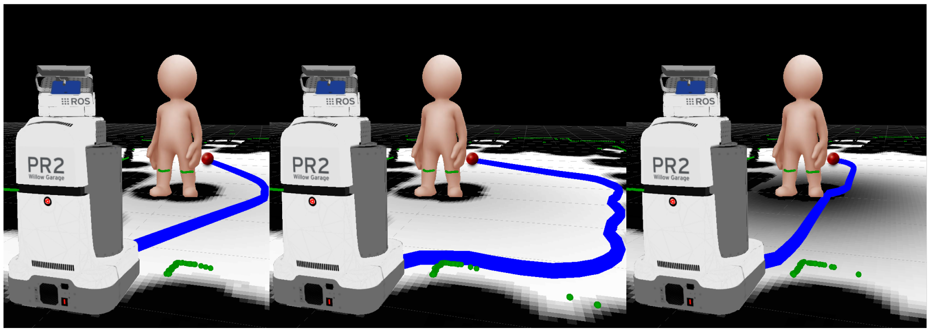


Fig. 4: Results using ROS Navigation, $\sigma = 1.75$, and $P/A = \{3/3, 3/60, 3/235\}$

7 Discussion

As robots require more and more social behaviors, it will be necessary to precisely design systems where the robot will navigate in a particular way. With this in mind, it is worth considering the placid assumption that the Gaussian costmap addition is the ideal tool for the job. In particular, the existence of the large dead zone where small changes to the parameters will have no effect on the path and the discontinuities when configuring just one parameter, together present numerous challenges to tackle when configuring a system. Additional and more complicated cost functions may alleviate some of the problems. Figure 3c shows the heatmap when using the sum of *two* Gaussian functions. This particular example creates greater range for $\hat{y}$ for small variances, but has its own problems with discontinuities. This is clearly not the only other option, and we advise that designers of new cost functions use the techniques in this paper to test their new functions.

The choice of costmap approach is vitally important for social navigation. The Gaussian adjustment explored in this paper are certainly better than no costmap modifications. However, in tight environments like narrow hallways (as we encountered previously[8]), the optimal $\hat{y}$ for a given set of parameters may result in an invalid path that collides with other obstacles. Limiting the $\hat{y}$ value means only considering the smallest values of $\hat{y}/\sigma$ in Figure 2b, and thus the robot may end up taking the most direct path, moving uncomfortably close to the person. In accordance with the study of proxemics, a robot which mostly drives far away, but sometimes moves into a person's intimate personal space seems much worse than a robot that always drove a medium distance away. This led to our decision not to use Gaussians in that experiment.

As a final note, social robots need to take many metrics into consideration to find the paths that will be best suited for interacting with humans. We do not claim that the sole definition of "best" path relies upon just path length and closest distance to an obstacle. Finding the best cost functions, and planners that optimize a wide variety of metrics, will likely see much fruitful work in the coming years.

References

[1] Pieter Abbeel and Andrew Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *Proceedings of the 21st International Conference on Machine Learning (ICML)*, 2004.

[2] Pieter Abbeel, Dmitri Dolgov, Andrew Y. Ng, and Sebastian Thrun. Apprenticeship learning for motion planning with application to parking lot navigation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1083–1090, 2008.

[3] H. Choset, K.M. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L.E. Kavraki, and S. Thrun. *Principles of robot motion: theory, algorithms, and implementations*. MIT press, 2005.

[4] K. Dautenhahn, M. Walters, S. Woods, K.L. Koay, CL Nehaniv, A. Sisbot, R. Alami, and T. Siméon. How May I Serve You?: A Robot Companion Approaching a Seated Person in a Helping Context. In *Proceedings of the 1st ACM SIGCHI/SIGART conference on Human-robot interaction*, pages 172–179. ACM, 2006.

[5] D. Ferguson and M. Likhachev. Efficiently using cost maps for planning complex maneuvers. *Lab Papers (GRASP)*, page 20, 2008.

[6] R. Kirby, R. Simmons, and J. Forlizzi. COMPANION: A Constraint-Optimizing Method for Person-Acceptable Navigation. In *Proc. RO-MAN 2009*, pages 607–612, Toyama, Japan, 2009.

[7] Sergey Levine, Zoran Popović, and Vladlen Koltun. Feature construction for inverse reinforcement learning. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1342–1350, 2010.

[8] David V. Lu and William D. Smart. Towards More Efficient Navigation for Robots and Humans. IROS, 2013.

[9] J. Mainprice, E. Akin Sisbot, L. Jaillet, J. Cortés, R. Alami, and T. Siméon. Planning human-aware motions using a sampling-based costmap planner. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 5012–5017. IEEE, 2011.

[10] E. Marder-Eppstein, E. Berger, T. Foote, B. Gerkey, and K. Konolige. The office marathon: Robust navigation in an indoor office environment. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 300–307. IEEE, 2010.

[11] Andrew Y. Ng and Stuart J. Russell. Algorithms for inverse reinforcement learning. In *Proceedings of the 17th International Conference on Machine Learning (ICML)*, pages 663–670, 2000.

[12] L. Scandolo and T. Fraichard. An anthropomorphic navigation scheme for dynamic scenarios. In *Robotics and Automation (ICRA), 2011 IEEE International Conference on*, pages 809–814. IEEE, 2011.

[13] E.A. Sisbot, L.F. Marin-Urias, R. Alami, and T. Simeon. A human aware mobile robot motion planner. *Robotics, IEEE Transactions on*, 23(5):874–883, 2007.

[14] M. Svenstrup, S. Tranberg, H.J. Andersen, and T. Bak. Pose estimation and adaptive robot behaviour for human-robot interaction. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 3571–3576. IEEE, 2009.

[15] M. Svenstrup, T. Bak, and H.J. Andersen. Trajectory planning for robots in dynamic human environments. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 4293–4298. IEEE, 2010.