# NavDog: Robotic Navigation Guide Dog via Model Predictive Control and Human-Robot Modeling

Liyang Wang
Baidu Research Institute, Baidu USA
Sunnyvale, California
liyangwang@baidu.com

Jinxin Zhao
Baidu Research Institute, Baidu USA
Sunnyvale, California
jinxinzhao@baidu.com

Liangjun Zhang
Baidu Research Institute, Baidu USA
Sunnyvale, California
liangjunzhang@baidu.com

## ABSTRACT

Guide dogs can vastly improve vision-impaired people's daily-life quality by guiding them to destinations while avoiding obstacles. Animal guide dogs are costly for training. This paper presents a robot guide dog system to take a vision-impaired user to a destination while avoiding obstacles in the environment for both the user and the robot dog. A novel human-robot kinematic model and an MPC-based motion planning and control algorithm are proposed. We implement the method on a wheeled ground robot. All the sensors are mounted on the robot, and the human user does not have to take additional sensor devices. Simulation and real-world experiment results show that the proposed method can tackle challenging navigation tasks in narrow corridors for vision-impaired people.

## KEYWORDS

Mobile robot, navigation, service robot, optimal control, multi-agent system modeling

**ACM Reference Format:**
Liyang Wang, Jinxin Zhao, and Liangjun Zhang. 2021. NavDog: Robotic Navigation Guide Dog, via Model Predictive Control and Human-Robot Modeling. In *Proceedings of ACM SAC Conference (SAC'21).* ACM, New York, NY, USA, Article 4, 4 pages. https://doi.org/10.1145/3412841.3442098

## 1 INTRODUCTION

Applying modern technology to navigation assistance devices has great potential to assist vision-impaired people and improve their life quality. Current developments focus on three main categories, such as wearable devices [14], intelligent cane [9], and robot guide dogs [3, 6, 7, 10]. This article explores the robot guide dog solution.

In recent years, a large number of works related to robot guide dogs have been proposed. In [3], a learning method is used for the robot to recognize and follow the existing trail line better. In [10], the presented robot guide dog can generate multiple paths, store them, and retrace the desired stored path based on environmental observation. In [6], a robot platform is designed along with a navigation algorithm, which can lead vision-impaired people to the destination while avoiding obstacles. Despite the advances, these approaches mainly consider robot geometry and dynamics constraints of itself and assume the user always follows the same path as the robot, which generally does not hold. Indeed, the user follows the end of the rod that is connecting the robot and the human. Only making sure the robot guide dog is collision-free does not necessarily mean humans will also be collision-free. However, the combined human and robot system must be considered and modeled for the collision avoidance purpose. As compared with the robot itself modeling, less research has focused on the entire human and robot system modeling. In [7], a hybrid system modeling of a four-legged robot and human system is presented. This result mainly focuses on controlling the legged robot, and the proposed algorithm difficult for real-time implementation.

In this paper, a generalized kinematic human-robot model is proposed and a formulation of collision-free constraints using Model Predictive Control (MPC) is presented for robotic guide dog navigation. We focus on tackling the challenge of human-robot system modeling and collision-free optimal control. The formulated problem is solved based on sequential convex optimization, and Model Predictive Control (MPC)-based control algorithm is designed for navigation and obstacle avoidance. MPC technique has been proven effective and is increasingly used in robotics [2, 11–13, 18]. In this work, the cost function and constraints in MPC are designed to guarantee collision-free for the human-robot system and fast computation.

We implement our approach and test it in both simulations and real-world experiments using a wheeled ground robot. Experimental results show that the proposed methods can tackle challenging navigation tasks with sharp turns while ensuring collision-free for the vision-impaired user. In this paper's simulation environment, the algorithm with the commonly used bicycle model failed to make the robot pass a narrow aisle with a sharp turn. In contrast, the MPC algorithm with the proposed model can successfully control the robot to pass. We define modeling error as the difference between model prediction result and pose estimation result. From the calculation, the maximum modeling error is $0.15m$. The mean error is $0.057m$, and the error standard deviation is $0.031m$.

## 2 MODELLING AND MPC DESIGN

### 2.1 Human and Robot Connection

We refer to the work presented by Kaveh et al. [7], where the user and robot are connected with a rigid rod, with a free joint at both the human end and the robot end. For vision-impaired users, navigation information is transmitted through the rod. The human user is assumed to move so that she/he would face the guide dog along the connecting rod and keep approximately the same distance to the guide dog as the rod length. Assuming the human user will


*SAC'21, March 22-March 26, 2021, Gwangju, South Korea*

ACM ISBN 978-1-4503-8104-8/21/03…$15.00
https://doi.org/10.1145/3412841.3442098

always follow the rod end, we assume the force applied to the robot through the rod by the human user can be fully compensated. Thus the robot movement is only driven by its own control commands.

## 2.2 System Kinematic Model

Intuitively, the described connected human and robot system is similar to a vehicle. Hence a large body of work applies the bicycle model [8] when solving the motion planning problem. For the bicycle model, usually, the no-slip condition [5] is stressed. However, the system of guide-dog and humans does not follow such an assumption. Human movement can be pure rotation without position change while the robot is moving around the human. Another drawback of the bicycle model is limiting the ability to do sharp turns, which is rather common in indoor situations. Therefore, this section derives a novel human and robot kinematic model based on the relationship defined in Sec. 2.1.

*2.2.1 Robot State Update.* We denote the robot state as $[x^b, y^b, \theta^b]$, where $(x^b, y^b)$ represent the robot coordination and $\theta^b$ is the robot heading angle. For simplification, We select the unicycle model [4] to describe the robot movement. The kinematic model of an unicycle type robot in discrete time is described as:

$$\begin{cases} x^b_{k+1} = x^b_k + \Delta t \cdot \cos\theta^b_k \cdot v_k \\ y^b_{k+1} = y^b_k + \Delta t \cdot \sin\theta^b_k \cdot v_k \\ \theta^b_{k+1} = \theta^b_k + \Delta t \cdot \omega_k \end{cases} \tag{1}$$

Where $v$ is the linear velocity input, $\omega$ is the angular velocity input, $k$ represent for iteration step, and $\Delta t$ is the period of the update loop.

*2.2.2 Human state update.* We denote human position as $[x^h, y^h]$. Without assuming human speed is equal to the robot speed, the human velocity dynamics is challenging to formulate since both interactions between the robot and the human's intention would affect the velocity. However, since the human will follow the rod end and keep a constant distance from the robot, we can infer the human position at the next step based on the current configuration.

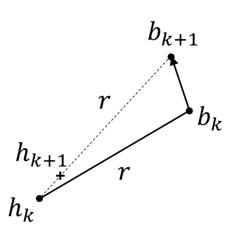



**Figure 1: Human position inferring.**

In this section, we propose a model of human position update. As shown in Fig. 1, $h_k$ and $b_k$ are the positions of human and robot at step $k$, respectively. The length of the rod is $r$, which is fixed when the robot moves. At next time step $k+1$, the robot moves to $b_{k+1}$. The next human position $h_{k+1}$ is assumed to locate on the line connecting $h_k$ and $b_{k+1}$, with a distance of $r$ to the location of $b_{k+1}$. Based on this assumption, we can further formulate the state update equation for the human user as follows,

$$\begin{cases} x^h_{k+1} = \frac{r}{d} \cdot x^h_k + (1 - \frac{r}{d}) \cdot x^b_{k+1} \\ y^h_{k+1} = \frac{r}{d} \cdot y^h_k + (1 - \frac{r}{d}) \cdot y^b_{k+1} \end{cases} \tag{2}$$

where it is defined that $d = \|h_k - b_{k+1}\|$.

## 2.3 MPC Design

An MPC framework is leveraged here for local motion planning for the robot. The prediction is achieved by utilizing the system kinematic model presented. The cost function and constraints are designed to guarantee collision-free as well as fast computation. In particular, the collision-free constraints are linearized at each iteration. The cost function is built to penalize the error of the states and the control effort. Equality constraints are constructed based on the linearized state update equations. In contrast, inequality constraints are applied to regulate the upper and lower bounds of states and control inputs, considering obstacle avoidance. The optimization problem is then formulated as:

$$\begin{aligned} \min \quad & (x_N - x_F)^T P (x_N - x_F) + \\ & \sum_{k=0}^{N-1} (x_k - x_F)^T Q (x_k - x_F) + \sum_{k=0}^{N-1} u_k^T R u_k \\ \text{s.t.} \quad & \begin{cases} x_0 = x_S \\ x_{k+1} = A x_k + B u_k \\ x_{\min} \le x_k \le x_{\max} \\ u_{\min} \le u_k \le u_{\max} \\ C x_k \le z \end{cases} \end{aligned} \tag{3}$$

In Eq. (3), $N$ is the prediction horizon, $x_S$ is the initial state. $x_{k+1} = Ax_k + Bu_k$ is the state space representation of the kinematic model. The state vector is defined as $x = \begin{bmatrix} x^b & y^b & x^h & y^h & \theta^b \end{bmatrix}^T$, and the control vector is defined as $u = \begin{bmatrix} v & \omega \end{bmatrix}^T$. The inequality constraints $x_{\min} \le x_k \le x_{\max}$, $Cx_k \le z$ and the local target state $x_F$ are carefully selected based on local occupancy grid map with a safe clearance, which makes the collision avoidance problem linear, convenient and fast to solve.

# 3 EXPERIMENTS

## 3.1 NavDog System Design

As shown in Fig. 2, there are three modules of our NavDog system: perception, global planning, and MPC-based motion planning and control. The perception module takes the raw sensor data, including wheel odometry, Lidar, and IMU, as inputs. It detects the obstacles' and the user's states based on the real-time Lidar data and calculates the robot state using Adaptive Monte Carlo Localization (AMCL) [16, 17]. The global planning algorithm is based on a graph search method. Inputs are the pre-build LIDAR environment map, the detected obstacles, the system state, and the destination information. The output is a global reference trajectory, which is also the input of MPC controller. The MPC controller relies on a linear QP solver called 'OSQP' [15].

The robot platform used is TurtleBot3 Waffle Pi [1], a ground wheeled robot. We take advantage an extended Kalman Filter (EKF) to filter the localization outputs. The EKF is realized with the OpenCV library. The system runs on ROS kinetic with Ubuntu

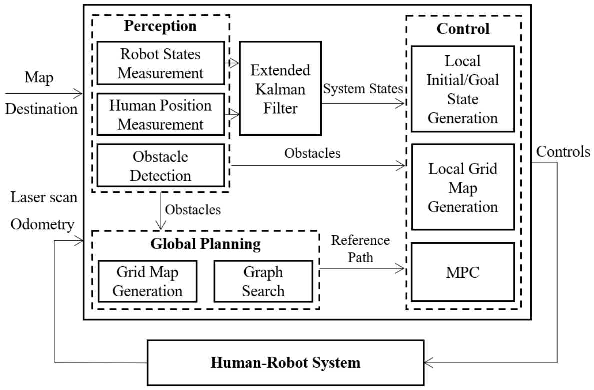



**Figure 2: NavDog System diagram.**

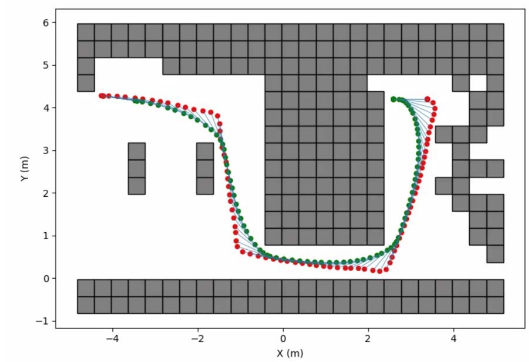


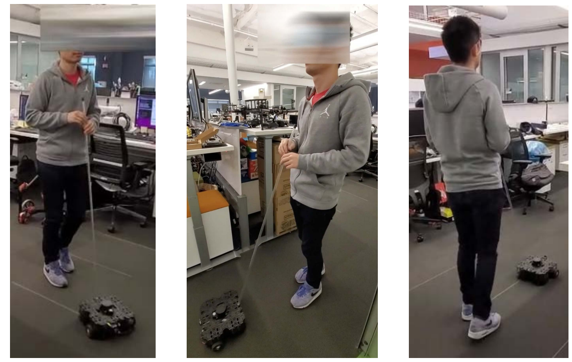

**Figure 3: Guide human in a simulation scenario. The robot (red) and the human (green) need to move from one room to another via a narrow corridor. The human and the robot are connected with a rod with fixed length (blue line). Start point is located at up right corner, and the end point is located at up left corner.**

16.04. Gazebo 3D rigid body simulator in ROS is exploited for simulation. A virtual environment is constructed, and Gazebo simulates and outputs the odometry and laser scan data. For visualization, RVIZ graphical interface in ROS is used to display the results. In the experiments, the localization maps are built with the ROS package ‘gmapping’, and the resolution of the maps is set as $0.05m$. The grid mask size is $0.4m$. The length of the rod connecting the human user and the robot is $0.8m$.

## 3.2 Guiding Human in Simulation

In the simulation, we built an area with two rooms as the test environment. We compared our proposed model with the bicycle model when implementing the algorithm.

First, we show the experimental results on the scenario where the human and robot system needs to move from one room to another via a narrow corridor (Fig. 3). The experiment results show that the robot can guide the human user to make a sharp turn, go through the narrow corridor, and successfully reach the destination. There is no collision for both robot and the human user.



(a) Image 1. (b) Image 2. (c) Image 3.

**Figure 4: Real world experiment set up.**

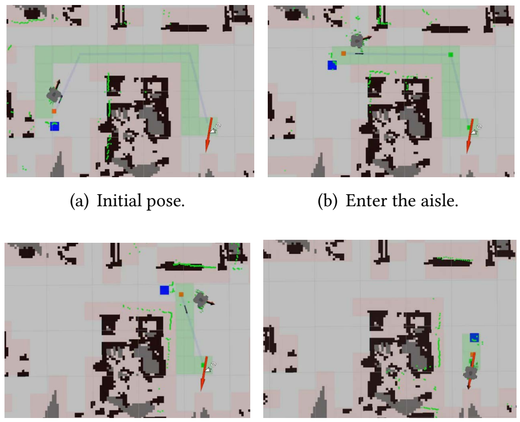
To further validate the kinematic human-robot model proposed in the paper, we also compare it with the bicycle model. We set up the experiment using the same scenario and change the proposed human-robot model to a bicycle model in the MPC controller. In this experiment, the system failed at the first sharp turn, located at position $(2.4, 0.5)$ and reported it could not find a feasible collision-free solution to turn. This result suggests that the proposed human-robot model can handle sharper turns compared with the bicycle model.

## 3.3 Guiding Human in Real World

In real-world experiments, a human user’s eyes are fully covered to simulate a vision-impaired user. We select an office area and build the corresponding map. The chosen experiment location includes two open space connected with a narrow aisle around $1m$ wide. The human-robot system’s initial location and navigation destination are located in two different open spaces, respectively. The planned path will pass through the aisle, and the trajectory shapes like the letter ‘U’. A plastic rod is used to connect the human user and the robot. During the test, the user’s eyes are covered. Hence the human guidance information is sensed from the rod only. Fig. 4 shows the real-world experiment set up.

Figure. 5 shows the experiment results, which are visualized using RVIZ. Same as the simulation test, the human position is marked with a blue square, and the system center is marked with an orange dot. The start pose is shown in Fig. 5(a), then enter the aisle (shown in Fig. 5(b)), exit the aisle (shown in Fig. 5(c)), and reach the destination successfully (shown in Fig. 5(d)).

We further analyze the accuracy of our human position modeling. We attach markers for tracking the human trajectory during an experiment and measure the marker positions as the ground truth. Fig. 6 shows human positioning results. In Fig. 6, the red line shows the ground truth human path; the blue line shows the estimated human path from EKF; the dashed green line shows the human position modeling and the dashed black line shows the human position measurements. The results suggest that the proposed human positioning method can track human position successfully. Then we define modeling error as the difference between model prediction result and pose estimation result. The maximum modeling error is



(a) Initial pose. (b) Enter the aisle.

(c) Exit the aisle. (d) Pose at end.

**Figure 5: Real world experiment (RVIZ screenshots).**

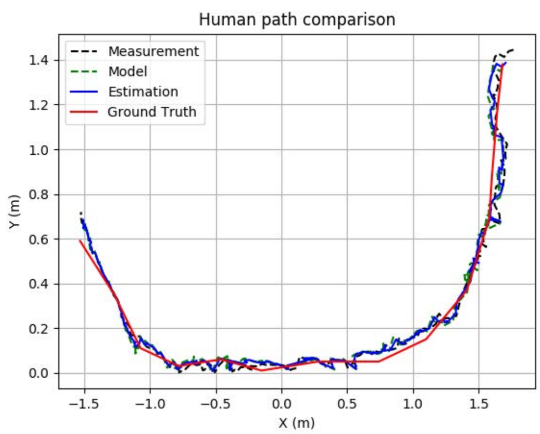



**Figure 6: Human positioning results.**

0.15$m$. The mean error is 0.057$m$, and the error standard deviation is 0.031$m$.

## 4 CONCLUSIONS

This paper proposes a novel approach to model the coupled system of the human user and the robotic guide dog. Based on such a model, an MPC motion planning algorithm is designed for navigation and obstacle avoidance. The formulation of the collision-free constraints leverages sequential convex optimization, and the method is improved in this article to ensure fast computation for real-time implementation. Complete system design is discussed afterward with human and robot localization and global path planning modules. To assess the system performance, both simulation and real-world experiments are performed. According to the experimental results, the proposed human-robot kinematic modeling method is validated, and the obstacle avoidance constraints for MPC controller design are proved to be effective.

Currently, the test is conducted with Turtlebot3 (wheeled robot), and the collision-free constraints are linearized in the sub-convex set. In the future, for robot sensitive to external force, dynamic modeling for the human-robot relationship will be developed. For the MPC controller design, quadratic or non-convex constraints will be formulated, and the crowd scenario will be studied to enable the system to work in more complex environments.

## REFERENCES

[1] Robin Amsters and Peter Slaets. 2019. Turtlebot 3 as a Robotics Education Platform. In *International Conference on Robotics and Education RiE 2017*. Springer, 170–181.

[2] Federico Augugliaro, Angela P Schoellig, and Raffaello D'Andrea. 2012. Generation of collision-free trajectories for a quadrocopter fleet: A sequential convex programming approach. In *2012 IEEE/RSJ international conference on Intelligent Robots and Systems*. IEEE, 1917–1922.

[3] Tzu-Kuan Chuang, Ni-Ching Lin, Jih-Shi Chen, Chen-Hao Hung, Yi-Wei Huang, Chunchih Tengl, Haikun Huang, Lap-Fai Yu, Laura Giarré, and Hsueh-Cheng Wang. 2018. Deep trail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired-learning from virtual and real worlds. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1–7.

[4] Alessandro De Luca, Giuseppe Oriolo, and Marilena Vendittelli. 2001. Control of wheeled mobile robots: An experimental overview. In *Ramsete*. Springer, 181–226.

[5] Gowtham Garimella, Joseph Funke, Chuang Wang, and Marin Kobilarov. 2017. Neural network modeling for steering control of an autonomous vehicle. In *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2609–2615.

[6] João Guerreiro, Daisuke Sato, Saki Asakawa, Huixu Dong, Kris M Kitani, and Chieko Asakawa. 2019. CaBot: Designing and Evaluating an Autonomous Navigation Robot for Blind People. In *The 21st International ACM SIGACCESS Conference on Computers and Accessibility*. 68–82.

[7] Kaveh Akbari Hamed, Vinay R Kamidi, Wen-Loong Ma, Alexander Leonessa, and Aaron D Ames. 2019. Hierarchical and Safe Motion Control for Cooperative Locomotion of Robotic Guide Dogs and Humans: A Hybrid Systems Approach. *IEEE Robotics and Automation Letters* 5, 1 (2019), 56–63.

[8] Jason Kong, Mark Pfeiffer, Georg Schildbach, and Francesco Borrelli. 2015. Kinematic and dynamic vehicle models for autonomous driving control design. In *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 1094–1099.

[9] Ayad Esho Korial and Mohammed Najm Abdullah. 2016. Indoor navigation for visually impaired/blind people using smart cane and mobile phone: Experimental work. *Journal of Information Engineering and Applications* 6, 5 (2016), 31–40.

[10] Rajesh Kannan Megalingam, Souraj Vishnu, Vishnu Sasikumar, and Sajikumar Sreekumar. 2019. Autonomous path guiding robot for visually impaired people. In *Cognitive Informatics and Soft Computing*. Springer, 257–266.

[11] Nikolce Murgovski, Gabriel Rodrigues de Campos, and Jonas Sjöberg. 2015. Convex modeling of conflict resolution at traffic intersections. In *2015 54th IEEE conference on decision and control (CDC)*. IEEE, 4708–4713.

[12] Tiago P Nascimento, Carlos ET Dórea, and Luiz Marcos G Gonçalves. 2018. Nonholonomic mobile robots' trajectory tracking model predictive control: a survey. *Robotica* 36, 5 (2018), 676–696.

[13] John Schulman, Jonathan Ho, Alex X Lee, Ibrahim Awwal, Henry Bradlow, and Pieter Abbeel. 2013. Finding Locally Optimal, Collision-Free Trajectories with Sequential Convex Optimization.. In *Robotics: science and systems*, Vol. 9. Citeseer, 1–10.

[14] Walter CSS Simões and VF De Lucena. 2016. Blind user wearable audio assistance for indoor navigation based on visual markers and ultrasonic obstacle detection. In *2016 IEEE International Conference on Consumer Electronics (ICCE)*. IEEE, 60–63.

[15] B. Stellato, G. Banjac, P. Goulart, A. Bemporad, and S. Boyd. 2017. OSQP: An Operator Splitting Solver for Quadratic Programs. *ArXiv e-prints* (Nov. 2017). arXiv:math.OC/1711.08013

[16] Emil-Ioan Voisan, Bogdan Paulis, Radu-Emil Precup, and Florin Dragan. 2015. ROS-based robot navigation and human interaction in indoor environment. In *2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics*. IEEE, 31–36.

[17] Lei Zhang, Rene Zapata, and Pascal Lepinay. 2012. Self-adaptive Monte Carlo localization for mobile robots using range finders. *Robotica* 30, 2 (2012), 229–244.

[18] Xiaojing Zhang, Alexander Liniger, and Francesco Borrelli. 2017. Optimization-based collision avoidance. *arXiv preprint arXiv:1711.03449* (2017).