A Smooth Trajectory Planning for a Semi Autonomous Wheelchair Using Particle Swarm Optimization

Ngo Van Thuyen

University of Technology and Education Ho Chi Minh City, Vietnam

01 Vo Van Ngan, Linh Chieu, Thu Duc, HCMC

Received: January 10, 2017; accepted: June 9, 2017

Abstract

This paper presented a motion planning algorithm for a semi autonomous wheelchair for severely disable people. The D was used to determine an optimal path to its goal position. An optimal motion planning using particle swarm optimization (PSO) to navigate the wheelchair to its global goal. The D* search guarantees a global path from the start position to the goal if there is any while the particle swarm optimization search provides a smooth movement for the wheelchair from one step to another. The simulation in Player/Stage showed that the wheelchair could find an optimal path and the planned trajectory was smoother compared with the motion planner using potential field only.*

Keywords: Semi-autonomous wheelchair, Particle swarm optimization, Motion planning

1. Introduction

Smart electric wheelchairs have been researched at several robotics and biomedical labs in the world. Some typical smart wheelchairs were introduced in [1-8]. Research on smart wheelchairs can be grouped into different categories: human – machine interface, sensors to sense the environment, modes of operation, and maps and landmarks for the wheelchairs.

A few smart wheelchairs operate like an autonomous robot, i.e., the users just need to input the desired position they want to reach and the wheelchair will plan and move to the goal. In this case, the wheelchair motion planner needs to have a complete map of the environment or it has to follow the black/white course on the floor, hence it cannot react to unknown obstacles or move in an unknown environment. This wheelchair is only suitable for severely disable people who cannot steer the wheelchair with joystick to follow a desired trajectory or for those who travel frequently in a known environment. Some semi-autonomous wheelchairs only support the users in obstacle avoidance; the user has to control the wheelchair hence this one does not need to know the environment.

This paper introduces a motion planning method for the wheelchair for severely disable people. The wheelchair is assumed to be equipped with relevant sensors and algorithm to detect any obstacle on its course and to localize itself in a working environment, i.e., it can determine its current position with an acceptable error. The motion planning algorithm for a semi autonomous wheelchair is shown in Figure 1. We assume that the wheelchair moves in a partially known environment and knows its initial location either by a localization algorithm or provided by the user. The desired location is inputted by the user, for example, the user may request the robot to move to the kitchen, the dinner table or the sink, etc. D* algorithm is performed to find a path for the wheelchair. If a path is found in this step, it is optimal subject to the defined heuristics and the path for the reference point is always safe. Once the path has been found, a motion planning algorithm using PSO provides the control inputs (v, w) for the wheelchair to move. When the wheelchair moves along its desired trajectory, if it detects any change in the environment, e.g., an unexpected obstacle on its course, it will re-plan its path and follows the trajectory produced by PSO motion planning algorithm.

The paper is organized as follows. In section 2, path planning using D* algorithm is presented, the motion planning using PSO is discussed in Section 3. Simulation results are provided in section 4. A conclusion and future work are given in Section 5.

2. Path planning

For a wheelchair to move to its desired position, a global path planning is needed to ensure the wheelchair could travel to its goal if there exists such a path. The D* path planning algorithm [10] is used to plan an optimal path for the wheelchair to move to its goal. Similar to A* search which was widely used for mobile robot planning, D* path planning provides an optimal path with minimum cost, however, it also allows a fast re-planning in a dynamic environment

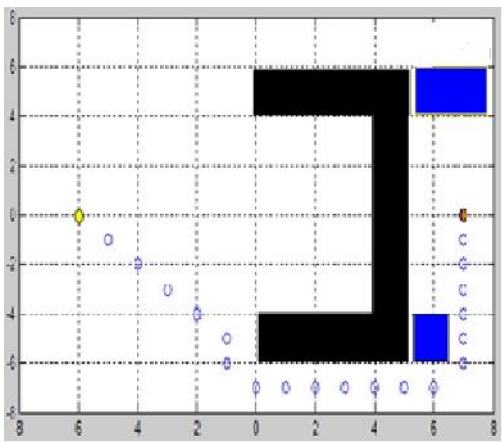
* Corresponding author: Tel.: (+84) 935.34.22.23
Email: thuyen.ngo@hcmute.edu.vn

where there maybe unexpected obstacles or the wheelchair has limited information about its working environment from the beginning. The heuristic cost in this case is the length of the path. In particular cases, the cost can include the distance of the planned path to the obstacle to ensure a safe movement. Fig. 1 shows the optimal planned path for a wheelchair to move from its initial location positioned at (0,-6m) to its desired location at (0,7m) in 2D Cartesian coordinates in a working environment with 16mx16m in width. There were a known U-shape obstacle and two unknown obstacles.

3. Motion Planning Using Particle Swarm Optimization

The common kinematic model for a non-holonomic three-wheel wheelchair [9] is described as:

$$\begin{aligned} \dot{x}(t) &= v\cos\theta \\ \dot{y}(t) &= v\sin\theta \\ \dot{\theta}(t) &= \omega , \end{aligned} \tag{1}$$



Notes:

: Unknown obstacle

: Known obstacle

: way-points

Fig. 1. The D* planned path for a wheelchair with U-shape obstacle

where $x(t)$ and $y(t)$ denote the position of the centre point on the wheel axis, $\theta(t)$ represents the orientation, and inputs v and ω are the translational and angular velocities respectively. The wheelchairs have three degrees of freedom, i.e. two positional degrees of freedom (x, y) and one orientational θ, which are related to a non-holonomic constraint equation implied in the model:

$$-\dot{x}\sin\theta + \dot{y}\cos\theta = 0 \tag{2}$$

The intuitive interpretation of this kinematic constraint equation is that the wheelchairs cannot move side-way but only forward and backward. In addition, the translational and angular velocities are limited by:

$$|v| \le v_{\max} \text{ and } |\omega| \le \omega_{\max} \tag{3}$$

Note that this model is applicable for many indoor wheeled robots while for outdoor vehicles, wheel-to-ground friction should be taken into account in the modeling.

In this paper, Particle Swarm Optimization [11] is used to find translational and angular velocities for the wheelchair to optimally move from one way-point to next way-point of the planned D* path with respect to a predefined fitness function. Fig. 2 shows four different trajectories for the wheelchair to move point A to point B. Generally, for the robot to move from point A to point B optimally, 3 factors have to be compromised: (i) The path has to be the shortest; (ii). The trajectory has to be smooth to make the user comfortable; (iii). The path is safe, i.e., the wheelchair does not collide with any obstacle.

One can see that, a set of translational and angular velocities can be found to move the wheelchair from A to B following the 2^{th} path which is the best path with respect to three factors mentioned earlier.

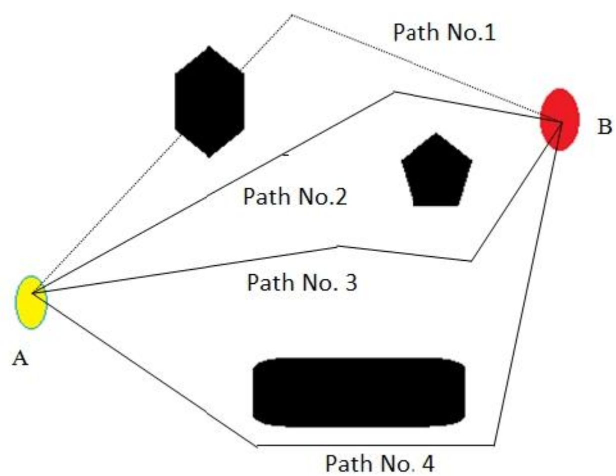


Fig. 2. The possible paths for the wheelchair to move from point A to point B

In Particle Swarm Optimization, a set of moving particles are initially thrown into the search space. Each particle, having a position and a velocity, knows its own position and the fitness function to evaluate its solution quality. Each particle randomly searches through the problem space by updating its own memory with its position and the social information gathered from other particles. At each time step, the movement of a particle is a compromise of three behaviors: to move randomly (its own way), to go toward its best previous position (memory), and to go toward its neighbor (global), best position. This evolutionary selection is described by the following equations for the i^{th} particle [11]:

$$V_i^{k+1} = V_i^k + c_1 r_1 \left(P_i^k - X_i^k\right) + c_2 r_2 x(P_g^k - X_i^k) \quad (4)$$

$$X_i^{k+1} = X_i^k + V_i^{k+1} \quad (5)$$

Where

X_i^k, X_i^{k+1}: Positions of i^{th} particle at k^{th} and $(k+1)^{th}$ iterations

V_i^k, V_i^{k+1}: Velocities of the i^{th} particle at k^{th} and $(k+1)^{th}$ iterations

c_1, c_2, r_1, r_2: Random numbers in the interval of [0,1]

P_i^k: Best position of i^{th}particle so far (local best)

P_g^k: Best position of the swarm so far (global best)

In this work, the solution is the translational and angular velocities v_i, ω_i, hence v_i, ω_i are encoded as the position of the Particle, i.e., $X_i^k = \{v_i, \omega_i\}$, $i = 1..N$ where N is the number of particles. The PSO algorithm to find the best solution (best v_i, ω_i) is as follow:

Step 1. Position initiation

N particles with random positions X_i^0 are initiated in the search space. Assume there are 100 particles, that means there are 100 pairs of random translational and angular velocities (v_i, ω_i).

Velocity limits: When the positions of particles are randomly initiated or calculated, their values can be optimal but they may be outside the velocity limits that the wheelchair can response. In order to avoid this situation, the positions of the particles will be limited to $\{v_m \ , \omega_m \}$. Those velocity limits depend on the dynamics of the wheelchair and the comfortable speed that the user may choose. In reality, the wheelchair must not move backward (i.e., $v < 0$) because that will cause a very discomfort feeling for the user. Hence, every position of the particles are subject to (6) and (7).

$$v_i(k) \leftarrow v_m \quad \text{if } v_i(k) > v_m \quad (6)$$

$$v_i(k) \leftarrow 0 \quad \text{if } v_i(k) < 0$$

$$\omega_i(k) \leftarrow \omega_m \quad \text{if } \omega_i(k) > \omega_m \quad (7)$$

$$\omega_i(k) \leftarrow -\omega_m \quad \text{if } \omega_i(k) < -\omega_m$$

Step 2. Calculation of fitness function

If the wheelchair moves with any pair of velocities initiated in Step 1, the next position of the wheelchair is estimated as (8):

$$\begin{bmatrix} x_i(k+1) \\ y_i(k+1) \\ \theta_i(k+1) \end{bmatrix} = \begin{bmatrix} x_i(k) \\ y_i(k) \\ \theta_i(k) \end{bmatrix} + \begin{bmatrix} \cos\theta_i(k) & 0 \\ \sin\theta_i(k) & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v_i(k) \\ \omega_i(k) \end{bmatrix} \Delta T \quad (8)$$

Where:

$x_i(k+1), y_i(k+1), \theta_i(k+1)$: Position and orientation of the wheelchair at iteration $(k+1)^{th}$,

$x_i(k), y_i(k), \theta_i(k)$: Position and orientation of the wheelchair at iteration k^{th}

$v_i(k), \omega_i(k)$: Velocities (control command) fed to the wheelchair at iteration k^{th}

ΔT: Sampling time.

As mentioned earlier, the goal of PSO is to find a set of $\{v, \omega\}$ so that the wheelchair can move from one way-point to the next one optimally in terms of path length, trajectory smoothness and safety.

Let d_f be the distance from the position of the wheelchair at $k+1$ when fed with $\left(v_i(k), \omega_i(k)\right)$ to the next way-point, $\Delta\theta_f$ the difference of wheelchair current orientation and the its next orientation at $k+1$ when fed with $v_i(k), \omega_i(k)$, d_f and $\Delta\theta_f$ are defined as follow:

$$d_f = \left(x_i(k+1) - x_w\right)^2 + \left(y_i(k+1) - y_w\right)^2 \quad (9)$$

$$\Delta\theta_f = \theta_i(k+1) - \theta_i(k)$$

Where $(x_w \ , y_w \ )$ is the position of the next way-point the wheelchair is moving to.

The fitness function for the i^{th} particle is defined as follow:

$$f_i = \alpha_1 d_f + \alpha_2 \left|\Delta\theta_f\right| + \alpha_3/\rho_i \quad (10)$$

Where:

ρ_i : Distance from the current position of the wheelchair to the nearest obstacle

$\alpha_1, \alpha_2, \alpha_3$: Weight of each factor, the bigger the value, the more importance of that factor appears in the fitness function.

With fitness function defined as in (10), the smaller the value of f_i, smoother, shorter and further from the obstacle the resulted path.

Step 3. Find the best position

This step is to find the best position of the swarm P_g^k, as well as the best position of each particle P_i^k. When initiated in Step 1, the local best of each particle is the initial position itself ($P_i^0 = X_i^0$) and the global best of the swarm is the position of the particle with smallest value of fitness function calculated as (10) ($P_g^0 = P_i^0$ with f_i).

When each fitness function is calculated for each particle, the local best for each particle and global best of the swarm is updated as follow:

$$P_i^{k+1} = \begin{cases} P_i^k & \text{if } f_i(P_i^k) < f_i(X_i^{k+1}) \text{ with } i = 1,..,N \\ P_i^{k+1} & \text{otherwise} \end{cases}$$

$$P_g^{k+1} = \begin{cases} P_g^k & \text{if } f_g(P_g^k) < f_i(X_i^{k+1}) \; i = 1,..,N \\ b \quad P_i^{k+1} & \text{otherwise} \end{cases}$$

Step 4. Position and velocity update

At each time step, the position and velocity of each particle are updated using equation (4) and (5).

4. Simulation result

The motion planning algorithm for the wheelchair is simulated in Player/Stage on Linux. The Player Project was developed by Gerkey, B., Vaughan, R., and Howard [12] which provided an advanced robotics simulation and interface. Player and Stage provide both the tools to simulate, as well as communicate to a physical robot system. Player allows user to interface with a variety of commercial mobile robots for educational purposes such as AmigoBot, Pioneer II… The working environment is 16mx16m in width and is devided into 289 cells (1mx1m). In the experiment, the motion planning using D* and PSO is compared with the one which uses potential field method to move between way-points and the translational velocity was kept constant.

In the first case, the wheelchair initial position is (-7, -6) m and it needs to move to position of (5, 7)m. The paths of the wheelchair when potential field and PSO are used are shown in Fig. 3 and Fig. 4, respectively.

Fig. 5 and Fig. 6 show the amplitude of the angular velocities of the wheelchair in both methods. The bigger fluctuation of the angular speed implies that the wheelchair changes its direction abruptly. One can see that, the motion planning using PSO allows the wheelchair to move a shorter and smoother path compared to the potential field method. In this case, the angular velocity of the wheelchair fluctuates strongly between (-1.4, 1.3) rad/s where as it fluctuates only between (-0.2, 0.4) rad/s when PSO motion planning is used. Visually, Figure 4 and 5 show how smooth the paths are in the two cases.

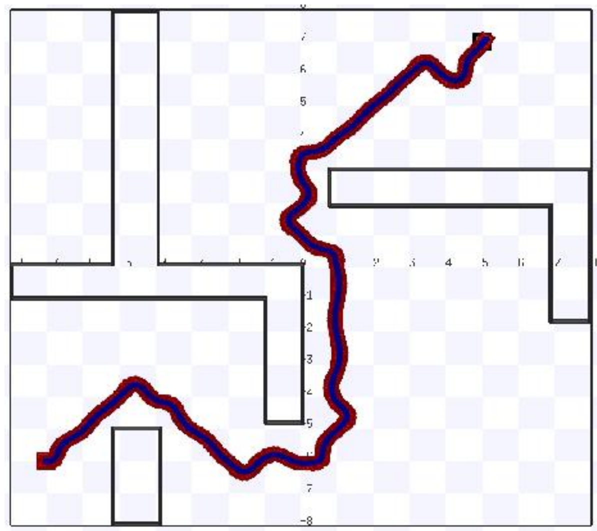


Fig. 3. Wheelchair path using potential field motion planning to follow the D* planned path

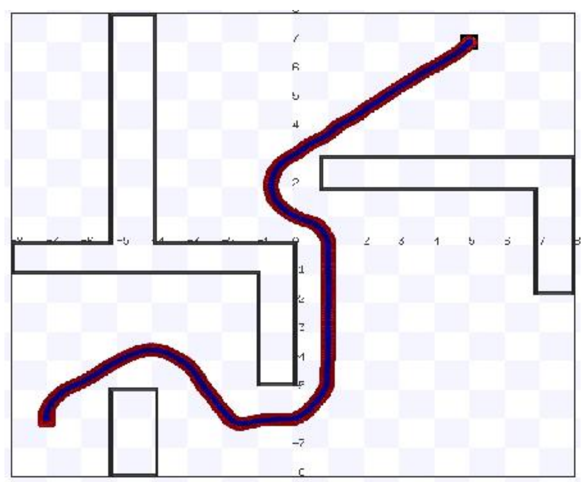


Fig. 4. Wheelchair path using PSO to follow the D* planned path

Fig. 7 to Fig. 9 show the simulation result on Player/Stage where the wheelchair has to move. In this scenario, there exists a U-shape obstacle in the environment. The wheelchair initial position is at (-6, 0)m and it needs to move to the position (0,7)m. It has been proved in [13] that, the wheelchair cannot navigate to its desired position in this case using potential field as a global motion planner.

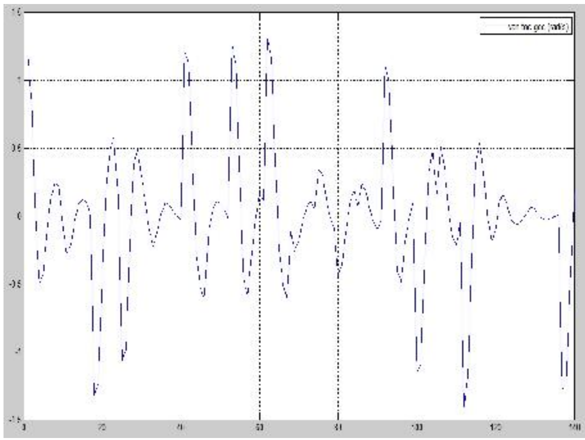


Fig. 5. Fluctuation of angular velocity (rad/s) over time (s) of wheelchair when using potential field motion planning to follow the D* planned path

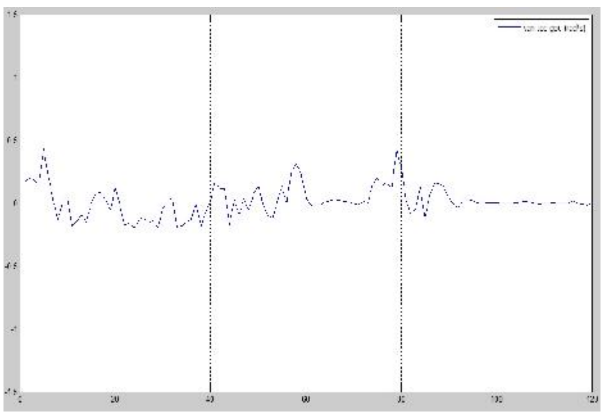


Fig. 6. Fluctuation of angular velocity (rad/s) over time (s) of wheelchair using PSO motion planning to follow the D* planned path

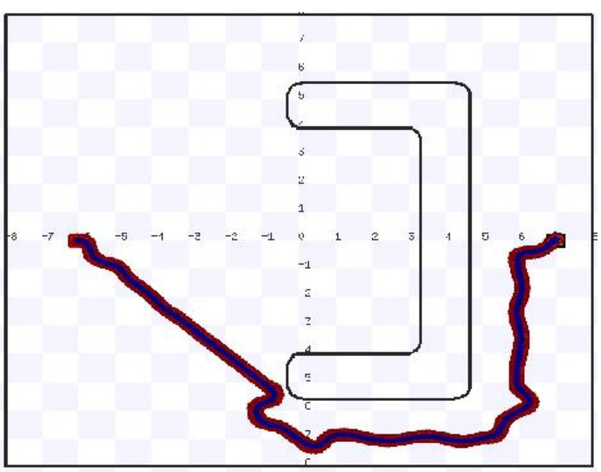


Fig. 7. Wheelchair path using potential field motion planning to follow the D* planned path with U-shape obstacle

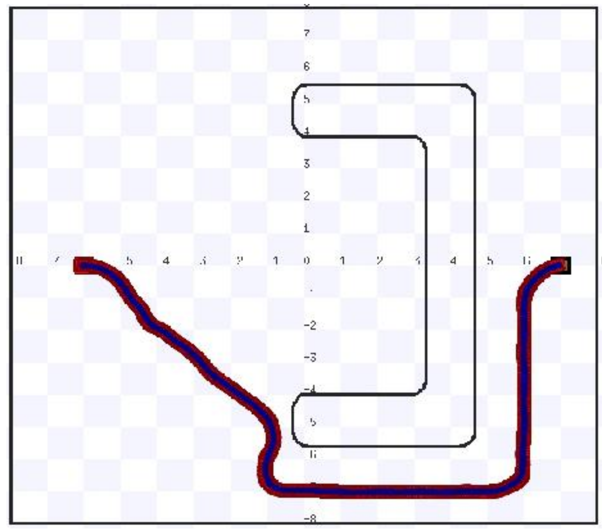


Fig. 8. Wheelchair path using PSO motion planning to follow the D* planned path with U-shape obstacle

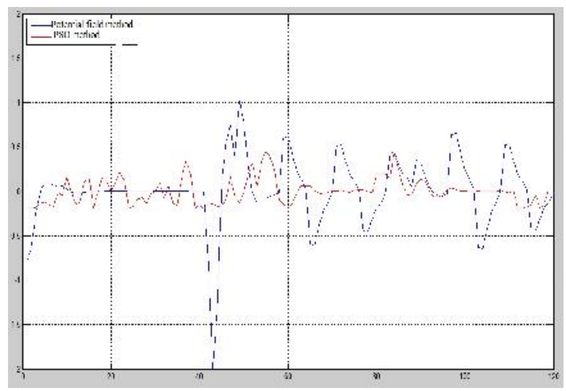


Fig. 9. Comparison of angular velocity fluctuations in potential field motion planning and PSO motion planning

However, with D* to find a global path, the potential field method can be used to navigate the wheelchair from one way-point to the next and eventually to its desired position as shown in Fig. 7. Fig. 8 shows the wheelchair path when PSO motion planning is used. Fig.9 shows the fluctuation of the angle velocity of the wheelchair in both cases.

The simulation results in two cases show that the wheelchair moves much more smoothly with the PSO motion planning than in the case of potential field.

5. Conclusion

In this work, we have presented a global motion planning for an autonomous wheelchair using D* search and Particle Swarm Optimization. The D* search allows fast re-planning in case the environment changes whereas the PSO motion planning proves to provide a smooth and short trajectory. The fitness function for the PSO motion planning comprises of three factors that needs to be minimized: the path length, the smoothness of the trajectory, and the safety of the trajectory. The simulation result shows that the PSO motion planning can provide a suitable motion for the wheelchair that can be used for severely disable people.

References

[1] http://www.ics.forth.gr/~tsakiris/Projects /grants.html, accessed on March 10th 2016

[2] Simpson RC, LoPresti EF, Hayashi S, Guo S, Ding D, Cooper RA., "Smart Power Assistance Module for manual wheelchairs. Technology and Disability: Research, Design, Practice and Policy", 26th International Annual Conference on Assistive Technology for People with Disabilities (RESNA) [CD-ROM]; 2003 Jun 19–23; Atlanta, GA. Arlington (VA); RESNA Press; 2003.

[3] Parikh SP, Rao RS, Jung SH, Kumar V, Ostrowski JP, Taylor CJ., "Human robot interaction and usability studies for a smart wheelchair", Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS); 2003 Oct 27–31; Las Vegas, NV. Piscataway (NJ), 2003. p. 3206–11.

[4] http://as.ms.t.kanazawa-u.ac.jp/as-e.html, accessed on May 14th 2016.

[5] G. E. Fabiani, D. J. McFarland, J. R. Wolpaw, and G. Pfurtscheller, "Conversion of EEG Activity into Cursor Movement by a Brain- computer Interface", Transactions on Neural Systems and Rehabilitation Engineering, vol. 12, pp. 331-338, 2004.

[6] D. J. Krusienski and J. J. Shih, "A Case Study on the Relation Between Electroencephalographic and Electrocorticographic Event-Related Potentials", in the 32nd Annual International Conference of the IEEE EMBS, 2010.

[7] D. Murray and J. J. Little, "Using Real-Time Stereo Vision for Mobile Robot Navigation", Autonomous Robots, vol. 8, pp. 161–171, 2000.

[8] D. J. McFarland and J. R. Wolpaw, "EEG-Based Communication and Control:Speed-Accuracy Relationships", Applied Psychophysiology and Biofeedback, vol. 28, pp. 217-231, 2003.

[9] Y. Yamamoto, and X. Yun, "Coordinating locomotion and manipulation of a mobile maniplator", in Recent trends in mobile robots, vol. Y.F. Zheng, Editor. 1993, World Scientfic: Singapor ; London, pp. 157-181.

[10] S. Anthony, "Optimal and Efficient Path Planning for Partially-Known Environments", Proceedings of the International Conference on Robotics and Automation: 3310–3317, 1994

[11] J. Kennedy and R. Eberhart, "Particle swarm optimization", in Proc IEEE International Conference on Neural Networks, Dec. 1995, pp. 1942-1948.

[12] B. Gerkey, R. Vaughan, and A. Howard, "The Player/Stage Project: Tools for Multi-Robot and Distributed Sensor Systems", Proceedings of the International Conference on Advanced Robotics 317-323, 2003.

[13] C. Sacramento, "Potential field methods and their inherent limitations for mobile robot navigation",. Robotics and Automation, IEEE International Conference on August 2002: p. 345-402.