Research on Cooperative Motion of Swarm Mobile Robots Based on PSO and Multibody System Dynamics

Peter Eberhard and Qirong Tang

Institute of Engineering and Computational Mechanics, University of Stuttgart
Pfaffenwaldring 9, 70569 Stuttgart, Germany
{eberhard, tang}@itm.uni-stuttgart.de

Keywords: Particle swarm optimization, Multibody system dynamics, Motion planning, Obstacle avoidance

1. Introduction

Swarm mobile robots can be used in many important scenarios. However, controlling a swarm robotic system is still a challenge which evokes our research interests. In this study, a mechanical model is built for swarm robots motion planning which is inspired by particle swarm optimization and combined with multibody system dynamics. It uses the augmented Lagrangian multiplier method to treat the constraints and an independent module to handle obstacle avoidance.

2. From basic PSO to mechanical PSO

The original PSO (Particle Swarm Optimization) was proposed by Kennedy and Eberhart in 1995 [1]. Some researchers modified or improved it then used it for many applications, e.g., for robotics as in [2]. The standard PSO model for all \( n \) particles can be described briefly by

\[
\begin{bmatrix}
    x_{k+1}^1 \\
    x_{k+1}^2 \\
    \vdots \\
    x_{k+1}^n
\end{bmatrix} =
\begin{bmatrix}
    x_k^1 \\
    x_k^2 \\
    \vdots \\
    x_k^n
\end{bmatrix} +
\begin{bmatrix}
    c_1 r_1^1 \left( \text{x}_{k+1}^\text{best} - x_k^1 \right) + c_2 r_2^1 \left( \text{x}_{k+1}^\text{swarm} - x_k^1 \right)
\end{bmatrix},
\]

for details and notation please see [3].

So far, many algorithms and methods are used for swarm robots motion planning, among them traditional and recursive biology inspired methods, see [3]. The PSO algorithm is very appealing due to its clear ideas, simple iteration equations, and the possibility to be mapped onto several robots or even swarm robots. Nevertheless, PSO itself (1) is only a general optimization algorithm and it needs several extensions for robot motion planning since robot always have mechanical properties, e.g., mass, volume, etc. In our study, the purpose to use PSO is robot motion planning thus their physical background must be considered, too. Meanwhile, we want to interpret the PSO algorithm as providing the required forces in the view of multibody system dynamics. Each particle (robot) is considered as one body in a multibody system which is influenced by forces and torques from other bodies in the system but without direct mechanical constraints between them. The forces are artificially created by corresponding drive controllers. On the other hand, derived from the Newton-Euler equations, the general form of equation of motion for swarm mobile robots can be formulated as

\[
M \ddot{x} + k = q \quad \text{or} \quad \ddot{x} = M^{-1}(q - k) = M^{-1}F.
\]

The force \( F \) is determined from three PSO-related parts which are defined as

\[
f_1^k = -h_j^k(x^k - \text{x}_{k+1}^\text{best}), \quad f_2^k = -h_j^k(x^k - \text{x}_{k+1}^\text{swarm}), \quad f_3^k = -h_j^k \omega.
\]

Furthermore, this model defines a state vector \( y = [x, \dot{x}]^T \), where \( x \) and \( \dot{x} \) are the translational and rotational position and velocity of the robots. Together with the initial conditions, first order of differential equation and (3), the motion of the swarm robots over time can be computed, e.g., by the simple Euler forward integration formula, which yields the mechanical PSO model

\[
\begin{bmatrix}
    x_{k+1}^1 \\
    x_{k+1}^2 \\
    \vdots \\
    x_{k+1}^n
\end{bmatrix} =
\begin{bmatrix}
    x_k^1 \\
    x_k^2 \\
    \vdots \\
    x_k^n
\end{bmatrix} +
\begin{bmatrix}
    (I_{na} - \Delta M^{-1} \dot{h}_j^k) \dot{x}_k^1 \\
    (I_{na} - \Delta M^{-1} \dot{h}_j^k) \dot{x}_k^2 \\
    \vdots \\
    (I_{na} - \Delta M^{-1} \dot{h}_j^k) \dot{x}_k^n
\end{bmatrix} +
\begin{bmatrix}
    M^{-1} h_j^k (\text{x}_{k+1}^\text{best} - x_k^1) + M^{-1} h_j^k (\text{x}_{k+1}^\text{swarm} - x_k^1)
\end{bmatrix},
\]

Comparing (1) and (4) one can see their corresponding relationships are

\[
\Delta t M^{-1} \dot{h}_j^k \leftrightarrow c_1 r_1^k, \quad \Delta t M^{-1} \dot{h}_j^k \leftrightarrow c_2 r_2^k, \quad \text{and} \quad I_{na} - \Delta t M^{-1} \dot{h}_j^k \leftrightarrow \omega,
\]

and we can introduce a sound mechanical interpretation to (5), for details see [3].
3. Constraint handling

Engineering optimization problems usually have constraints, e.g., if the robots (particles) are searching in the environment with some limitations. In this study, we also take into account the treatment of constraints. The general optimization problem with an objective function and constraints is

\[
\begin{align*}
\text{minimize} & \quad \psi(x) \\
\text{subject to} & \quad g(x) = 0, \quad m_e \text{ equality constraints}, \\
& \quad h(x) \leq 0, \quad m_i \text{ inequality constraints},
\end{align*}
\]

where \(x\) is the position of the particle bounded additionally by \(x_{\text{min}} \leq x \leq x_{\text{max}}\). For such an optimization problem, the augmented Lagrangian multiplier method can be used where each constraint violation is penalized separately by using a finite penalty factor. Thus, the minimization problem with constraints in (6) can be transformed into an unconstrained minimization problem as in [3], [4].

4. Platform, simulation and proposed real mobile robot

Developing a convenient user interface and simulation environment for the algorithms and model testing at initial stages is necessary. Based on the MATLAB GUI functions, an environment is developed which is shown in Figure 1. On this platform, we got some simulation results, see e.g. in Figure 2 a group of mobile robots searching in the environment with abilities of obstacle avoidance, mutual avoidance, acceleration/deceleration, steering etc. Figure 3 shows the Festo mobile robot platform ROBOTINO. We plan to use our mechanical PSO model on a group of such real robots in the following work.

References