Analysis of the PSO parameters for a robots positioning system in SSL

Marcos Aurelio Pchek Laureano [1,2], **Flavio Tonidandel** [2]

[1] Federal Institute of Parana, Curitiba – PR, Brazil

[2] University Center of FEI, São Bernardo do Campo – SP, Brazil

[1] marcos.laureano@ifpr.edu.br, [2] flaviot@fei.edu.br

The 23rd Annual RoboCup International Symposium 2019 — **RoboCup 2019**

Abstract

The changes in the Small Size League rules have brought greater possibilities of playing. With the increased complexity of soccer matches, the positioning of the robots has become important as a defense and attack mechanism. The learning of opposing team game playing has been shown to be effective, but an SSL soccer match indicates the need for solutions that analyze the strategy of the opposing team during the game and make any necessary adaptations. This paper proposes the use of the Particle Swarm Optimization (PSO) algorithm as an option to determine the positioning during the match. A prototype has been developed to validate the configuration parameters. Experiments in a simulator, analysis of game logs and results in a real matches have demonstrated the feasibility of applying the PSO algorithm to find the robots positions.

Particle Swarm Optimization

Particle swarm optimization (PSO) is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. It solves a problem by having a population of candidate solutions, here called particles, and moving these particles around in the search space according to the mathematical formula over the particles position and velocity.

PSO pbest and gbest equation

$$pbest(i,t) = min[f(P_i(k))], i \in 1,...,N_p, k = 1,...,t$$
$$gbest(t) = min[f(P_i(k))], i = 1,...,N_p, k = 1,...,t$$

PSO Velocity equation

$$V_i(t+1) = \underbrace{\underbrace{\omega V_i(t)}_{\text{inertia}}}_{\text{diversification}} + \underbrace{\underbrace{c_1 r_1 (pbest(i,t) - P_i(t))}_{\text{cognitive}} + \underbrace{c_2 r_2 (gbest(t) - P_i(t))}_{\text{social}}}_{\text{intensification}}$$

PSO Position equation

$$P_i(t+1) = P_i(t) + V_i(t+1)$$

Proposal

The first objective is to test, verify, and determine the configuration parameters (inertia, confidence, number of iterations and population size) of the PSO algorithm in order to optimize robot positioning in the field. The second is to demonstrate the effective positioning of robots based on the defense fitness function developed for this study.

To meet the defense requirements similar to those in human soccer, a fitness evaluation function is used in simulations applied to the defense position.

It evaluates four desirable situations for a defense formation:

- A minimum distance among the robots in order to make opponent's movements more difficult and decrease the opponent's chances to receive the ball, make passes or kick to goal;
- The view of the opponent's robots in relation to a certain point of interest is blocked;
- The view of the goal of all the opponent's robots is blocked by at least one robot of team, especially opponent robot with ball possession;
- Respect for the SSL rules on collisions between robots and invasion of the goal area.

General Equation

Field size is considered in centimeters and the *Goal* term can be any point of interest in the search space.

$$\begin{aligned} \text{fitness}(A, p(i,t), goal) = \; & f_{\text{MinDistance}}(A, p(i,t)) + \\ & f_{\text{CheckStraight}}(A, p(i,t), goal) + \\ & f_{\text{ProtectGoal}}(A, p(i,t)) + \\ & f_{\text{Colission}}(A, p(i,t)) + \\ & f_{\text{InvasionGoalArea}}(p(i,t)) \end{aligned}$$

Minimum distance

$$f_{\text{MinDistance}}(A, P) = \{$$
$$\text{distance}(p,a) = \sqrt{(p_x - a_x)^2 + (p_y - a_y)^2}$$
$$d = \text{distance}(p,a)$$
$$I_{c_d}(p,a) = \begin{cases} (1 - \frac{40}{d}), & \text{if } (1 - \frac{40}{d}) > 0 \\ \text{PMID}, & \text{otherwise} \end{cases}$$
$$\sum_{a \in A} \sum_{p \in P} I_{c_d}(p,a)$$
$$\}$$

Blocking the view of points of interest

$$f_{\text{CheckStraight}}(A, P, goal) = \{$$
$$\text{straight}(a,p,g) = \{((g_y - p_y) \times a_x + (p_x - g_x) \times a_y + (g_x \times p_y - p_x \times g_y))\}$$
$$s = \begin{cases} 0, & \text{if } a \text{ has the ball} \\ \text{PLOW}, & \text{otherwise} \end{cases}$$
$$I_{c_s}(p, a, goal) = \text{straight}(a, p, goal) + s$$
$$\sum_{a \in A} \sum_{p \in P} I_{c_s}(p, a, goal)$$
$$\}$$

Blocking the view of the goal

$$f_{\text{ProtectGoal}}(A, P) = \text{PMID} \times (\sum_{p \in P} \forall p \notin A_{ball})$$

Respect SSL rules

$$f_{\text{Colission}}(A, P) = \text{PHIGH} \times ((\sum_{a \in A} \sum_{p \in P} \text{pos}(a) = \text{pos}(p) + (\sum_{p \in P} \sum_{q \in P} \text{pos}(p) = \text{pos}(q) \ \wedge \ p_{id} \neq q_{id}))$$

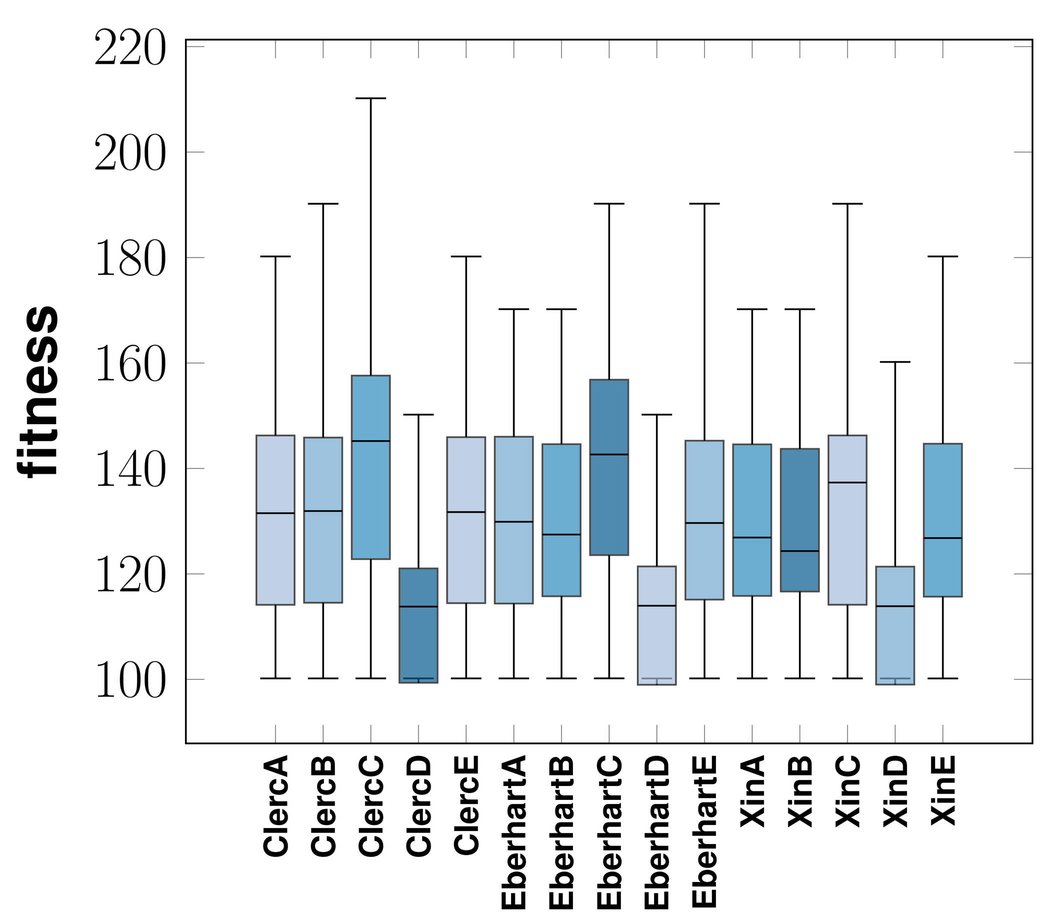
$$f_{\text{InvasionGoalArea}}(P) = \text{PHIGH} \times (\sum_{p \in P} \forall p \in \text{Goal area})$$

Discovery best PSO parameters

Inertia weight (ω) plays a key role in providing balance in the local and global exploitation process. $\omega = 0.7298$, $\omega = 0.5 + \frac{rand(0,1)}{2}$ and $\omega = (\omega_{start} - \omega_{end}) \times \frac{it_{Max} - it}{it_{Max}} + \omega_{end}$

Fifteen simulation scenarios to find parameters ω, c_1 and c_2. 15 seconds of movement to simulate a real game situation; The population size (from 50 to 200); Number of iterations (from 50 to 500); Global and local best neighborhood topologies were used.

Results



Scenario settings: a) $c_1 = 1$ and $c_2 = 2$; b) $c_1 = 2$ and $c_2 = 1$; c) $c_l = c_2 = 1$; d) $c_1 = c_2 = 2$ and e) $c_1 = c_2 = 1.496$.

All scenarios with $c_1 = c_2 = 2$ have the best results, followed by the scenario with $c_1 = c_2 = 1.496$.

Experiments

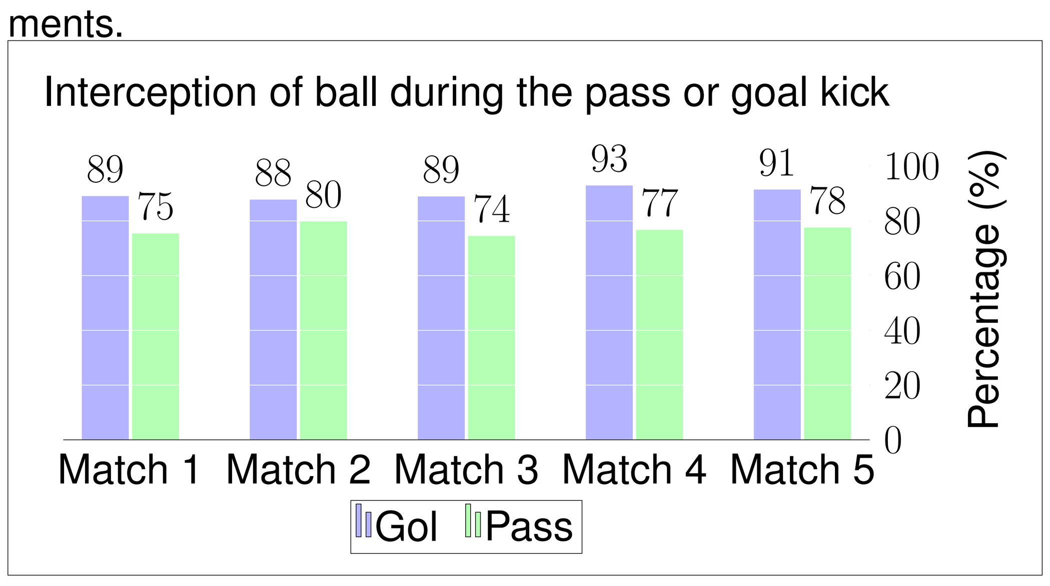
Three types of experiment were performed using PSO equations with parameters $\omega = 0.7298, c_1 = c_2 = 2$ to calculate the velocity and position of each robot that make up the particle, a grSim simulator, log analysis of RoboCup 2018 and the *Latin American Robotics Symposium* 2018 (LARS 2018) in five real matches.

Results – Simulator

In the grSim simulator, the ball and the opponent's robots were positioned differently for visual verification of the behavior of the team robots. These validations allowed us to identify and adjust some parameters of the fitness function.

Results – RoboCup 2018 playoffs

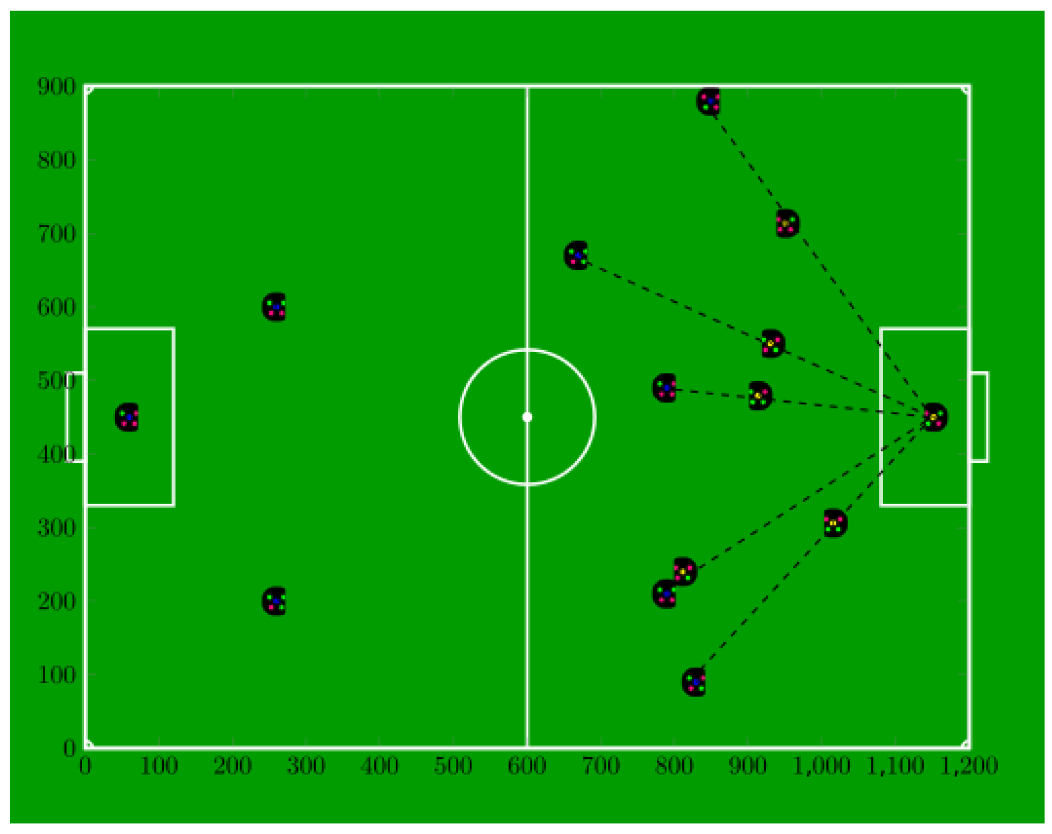
Five matches were selected from the RoboCup 2018 playoffs. Each scenario was evaluated 1,000 times with the parameters of the algorithm found in the previous experiments.



Results – LARS 2018 matches

These experiments allowed us to verify the positioning limitations caused by the time spent in processing and sending commands to the robots. In games with dead-ball situations, the positioning found by the system was always a difficult one for passes or kicks to goal.

Conclusions



Example of positioning

The experiments indicate that the best global neighborhood topology, the number of iterations (300), acceleration coefficients ($c_1 = c_2 = 2$), and the size of the population (100) meet the project requirements in terms of computational costs to be run during a real soccer match. For inertia, all the evaluated strategies were effective, and we adopted the value $\omega = 0.7298$.

The analysis of RoboCup playoff logs has demonstrated the effectiveness of the proposal for this paper. Experiments conducted during LARS 2018 have shown that it can be used to find field positioning during an real SSL soccer game, especially in games with dead-ball situations (e.g. indirect kicks). However, for a dynamic game, the movements of the opponent's must be considered in order to calculate the future positioning, which is the main challenge to be improved on future studies.

This study was financed in part by Instituto Federal do Paraná and the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001.