ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

EVOLUTIONARY MODEL FOR THE ITERATED N-PLAYERS PRISONERS' DILEMMA BASED ON PARTICLE SWARM OPTIMIZATION

SALLY ALMANASRA

[1]Assistant Professor, Arab Open University, Faculty of Computer Studies, Riyadh, Saudi Arabia

E-mail: s.almanasra@arabou.edu.sa

ABSTRACT

Evolving the cooperative behavior in Iterated N-Players Prisoners' Dilemma (INPPD) is studied over several evolutionary models. These models presented solutions for evolving cooperative behavior among INPPD players. Studying existing models revealed that when the number of the players' increases, the models lose their capabilities in maintaining stable levels of cooperation between the players. In this paper, we present an evolutionary model for enhancing the cooperation levels in large population of INPPD players. The model focuses on optimizing the communication topology of INPPD, as well as building a knowledge base to support players' future decisions based on the evolved knowledge gained from historical actions taken by different players. The presented communication topology along with the knowledge base present considerable support for the evolutionary Particle Swarm Optimization (PSO) algorithm to evolve the players' strategies. The results showed that the model could increase the cooperative rate among INPPD player and allow players to achieve higher payoffs against benchmark strategies

Keywords: *Prisoners' Dilemma, Game Theory, Communication Topology, Particle Swarm Optimization, Knowledge base*

1. INTRODUCTION

The prisoner's dilemma (PD) game is one of the more well-known two player games in the field of Game Theory. As our real-life decision making is not commonly based on a single strategy, an iterated version of PD is developed and is known as iterated prisoner's dilemma game (IPD). IPD is found to support decision makers with different viewpoints (i.e. strategies). IPD is also used to model emergent cooperative behaviors in selfish populations as in biology [1], sociology [2], psychology [3] and economics [4].

Understanding the evolution of cooperation in humans is a challenge for economists and biologists. Conditional cooperation is important mechanisms to prevent free riding in social dilemmas [5]. In large populations, we found that single individuals should have a substantial influence on their peers. Therefore, influencing peers should be oriented toward the best cooperative players in the population to evolve cooperative behavior between the players.

Evolving cooperation behavior in a large population is a complex problem in modeling Iterated n-Players Prisoner's Dilemma (INPPD) [6]. The evolution of a cooperation behavior in a given population requires the discovery of a strategy that a large group of opponents follow throughout the game. However, discovering the opponents' strategies requires the use of evolutionary algorithms that can search for the most optimal strategies among a large number of possibilities.

In this paper, we focus on avoiding premature players' convergence issue that evolves the cooperative behavior of INPPD players. Existing communication topologies allow players to communicate with a static set of neighbors. This may lead players in group (G1) to imitate the behavior of the best players in that particular group (G1), since the players in (G1) have no extra knowledge on the behavior of other players in other groups in that population. To solve this problem, we present an alternative topology that explores the behavior of wider range of players in the population. The new topology is dynamic to facilitate effective exchange of experiences between players in large populations and to ensure the provision of sufficient communication channels between players. Hence, particle swarm optimization (PSO) is used here to evolve players' strategies.

The remainder of the paper is organized as follows. Sections 2 and 3 provide an overview on INPPD and the PSO, respectively. Section 4 discusses the related works of evolving the cooperative behavior of INPPD. The proposed model is introduced in Section 5. A number of performance tests are conducted and the results are presented in Section 6. A concluding remark is given in Section 7.

2. ITERATED N-PLAYERS PRISONERS' DILEMMA

Two IPD players may play against each other several times. This allows players to generate strategies based on previous interactions. Hence, any given player's move has a substantial effect on the behavior of a future opponent's moves. The consequences of IPD eliminate the domination of single strategy of mutual defection, since players use complex strategies based on previously played moves. The purpose behind this concept is to maximize the payoff scored by a given player against other players. Dilemma occurs when players are unaware of each other's actions until the actions are taken. In that case, players are always cautious in performing cooperative actions, whereas other players are defecting [6],[8].

Assessing the performance of a given player depends on the payoff scored by that player in each game and generation. The concept is that the higher the payoff, the better is the strategy. As part of the rules of the game, the players should be aware of the payoff matrix being used throughout the game. Yao & Darwen [8] presented a payoff matrix to support INPPD as shown in Table 1 and Table 2. The columns and rows refer to the number of cooperators, and choices that a given player can make, respectively.

Table 1. Payoff matrix of INPPD

		Number of cooperators among the remaining n-1 players				
		0	**1**	**2**	**…**	***n*-1**
Player A	***Cooperate***	c_0	c_1	c_2	…	c_{n-1}
	Defect	d_0	d_1	d_2	…	d_{n-1}

Table 2: Numerical values of INPPD actions

		Number of cooperators among the remaining n-1 players				
		0	**1**	**2**	**…**	***n*-1**
Player A	***Cooperate***	0	2	4	…	2(*n*-1)
	Defect	1	3	5	…	2(*n*-1)+1

Each prisoner in INPPD represents a player and the players are distributed over a game space (usually a lattice). To facilitate the communication between players during the game, a set of communication topologies are available. These topologies specify the group of neighbors which are available for the center player to play against. Practically, the communication between the players is restricted to their pre-defined neighboring levels. The following sub-sections present a brief description on the well-known static topologies that are commonly used in INPPD [9], [10].

2.1 Ring Topology

In ring topology, each player is connected to its l immediate neighbors in a one-dimensional space (left-right or up-bottom). Therefore, each player can only share his experience with his immediate two neighbors. The overall impact of this topology on the population, defined as all participating players, may result in lower levels of cooperation because the players' knowledge is restricted to its own small group of neighbors.

Cooperative behavior in a given population of *n* players may emerge after several games, for example, in a tournament of *n* games. The low payoffs that a player gains after each game will motivate that player to change his behavior based on his own experience and the experiences of his own neighbors. Therefore, ring topology provides narrower levels of experience for each INPPD player.

2.2 Star Topology

Star topology is a fully connected structure that allows each player of the group to share information globally. This topology is mainly based on sharing the best experience, found by the best player, among all other players in the population. The neighborhood of a given player is the entire population. Thus, information is instantaneously distributed to all players, attracting the entire group to the superior behavior.

The star topology is one of the most efficient topologies in sharing experiences in small populations. However, star topology requires the processing of a huge amount of information as well as communication overhead in the information exchange between large numbers of players.

2.3 Von Neumann Topology

This structure adds another dimension to the searching algorithm in order to extend the ring communication structure. This structure considers immediate neighbors which are connected to a particular player from the left, right, up and bottom.

The Von Neumann topology is very useful for many optimization problems[9],[10].

The INPPD model proposed in [11] adopts the Von Neumann topology as a communication topology between INPPD players. The cells in the edges of the lattice have neighbors on the opposite sides of the lattice. However, the choice of the next player to interact with his neighbors during the simulation can be sequential or random.

2.4 Cluster Topology

In this topology, the players are divided into n clusters. Each cluster communicates directly with others through the connections, previously defined between players. Usually, each cluster is associated to a number of connections that are equal to the number of neighbor clusters.

The group is divided into three or four clusters. The players are connected to every other player in their cluster, but only a few connections between the clusters exist. The INPPD model presented in [6] shows that a high degree of community structure can ensure that cooperative players can insulate themselves from neighboring non-cooperating behaviors. Community structure is one form of clustering where collections of nodes are joined together in tightly knit groups between which only loose connections exist. The members of neighboring communities can update their behavior to imitate that of a more successful behavior and this technique proves that cooperation can propagate through society.

2.5 Random Topology

Given n players, n random symmetrical connections assigned between pairs of individuals exist. Random topology assigns connections at random between pairs of players. The lattice is formed in such a way that every player on the lattice has eight immediate neighboring players.

The work presented in [7] examines the effectiveness of co-evolutionary learning in a specialized environment with fixed and random communication structures. The consideration of random communication structures with inner and outer neighborhoods conducts the experiment. Inner neighborhood refers to the selection of group members within the eight immediate neighboring players, whereas the outer neighborhood refers to the selection of group members from anywhere across the entire population.

However, the communication topologies discussed in this section are varying in term of their efficiency in evolving cooperative behavior between INPPD players. In the next section, we discuss the current researches that focus on the communication topologies as tools for evolving cooperative behavior.

3. PARTICLE SWARM OPTIMIZATION

Swarm Intelligence (SI) is an attractive area that deals with the scalability issue for multi-agent systems while maintaining system robustness and individual simplicity. SI is an interesting computational technique that is inspired by the behavior of flocking, herding and insects swarming, where robust and coordinated group behavior requires a small set of simple local interactions between individuals, and between individuals and the environment[12],[13].

The swarm intelligent agents communicate and cooperate by implicit rules of cohesion, separation and alignment focusing to solve a problem which is guided by a decision metric known as fitness [14]. SI initiates several intelligent agents to solve a given optimization problem. These agents communicate and cooperate based on the corresponding SI approaches.

Particle Swarm Optimization (PSO) is the most popular approach in SI [14]. Like general evolutionary algorithms (EA), PSO maintains a population of individuals, which are called particles. These particles represent potential solutions to an optimization problem. However, PSO generates new individuals based on explicit mathematics models, instead of the well-known genetic operators (e.g. crossover and mutation) [15].

Searching for the best solution in PSO is carried out by endowing each particle to fly through the search space and update its velocity at regular intervals. This loop of updating continues towards both the best location it personally has found (personal best), and the globally best position found by the entire swarm (global best). The global best solutions are stored in a shared memory where all particles can access it to determine their individual velocities. However, PSO is the most favorable approach in terms of exploration and exploitation. The general flowchart of PSO is described in Fig. 1.

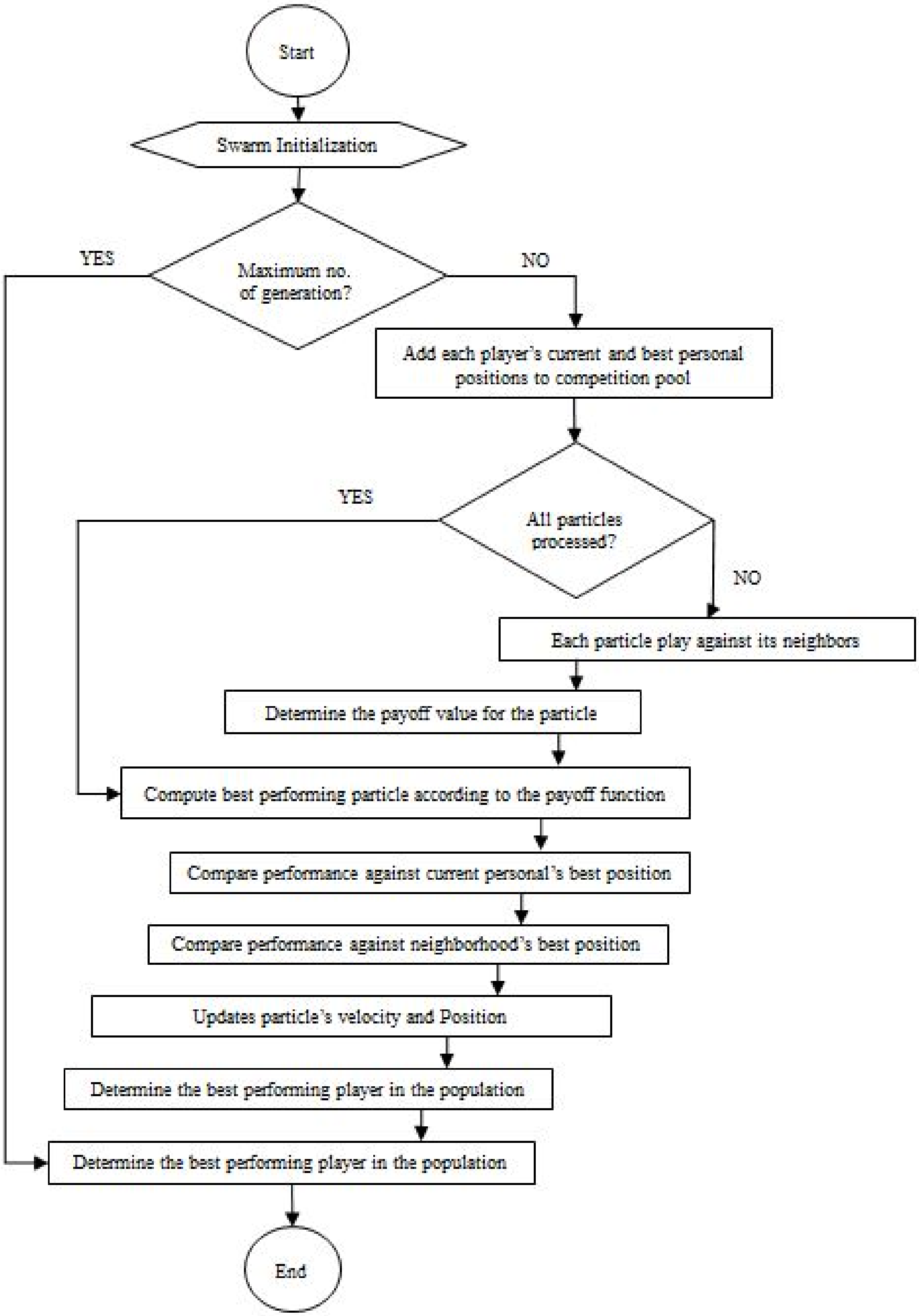

Figure 1: Flowchart of PSO Algorithm

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

4. RELATED WORKS

The evolution of cooperative behavior in INPPD is discussed in several studies. We found that some research presented models for evolving cooperative behaviors of the players based on: the different usage of the players' memory resource [16], the punishment in a structure population playing the social dilemma [5], the randomness of opponent memories sizes [17] and others.

The state of art reveals that many studies discuss different aspects for evolving cooperative behavior between INPPD players. Research shows that communication topologies between players have a great impact on the evolution of cooperative behavior [18]-[20].

The work presented in [18] focuses on understanding network-related factors that affect the evolutionary stability of a strategy. It is shown to be critical in making accurate predictions about the behaviour of a given player when reflected in a strategic decision making environment in the real-world. The goal of this work is to examine the effect of network topology on evolutionary stability. The results show that the topological connection patterns influence the decisions made by individual players over time.

In [20], the authors stress the role of the structure of the communication topologies in the play profile generated by their genetic algorithm-based model. The results show that communication topologies play a vital role in evolving cooperative behavior between the players.

The work published in [21] focuses mainly on the use of a graph concept to represent the communication topology between the players. The results showed that adjusting the degree of the graph has an impact on the cooperation rate in the population. Decreasing the degree of the graph will decrease the cooperation rate.

The work presented in [22] investigated the application of co-evolutionary training techniques based on PSO to evolve cooperation of INPPD. The experiments conducted in this work focuses on the impact of communication topology on the evolvement of cooperative behavior. The results revealed that fully-connected topology could enhance the cooperation ratio between the players compared to other communication topologies. However, the model presented in this work is oriented toward calculating the cost and benefits by adjusting the payoff matrix of INPPD and accept n choices rather than two exact choices with fixed probability.

In [23], the authors explore the dependence of the evolution of cooperation on soft control (well-designed updating rules) by an evolutionary IPD game. The shills (agents) are adopting their behavior based on the mechanism used by the particles in PSO. The results show that the cooperation can be promoted by the population of agents and that the frequency of the promotion can be enhanced if proper parameter settings are selected. This research also highlighted that adding agents to the population has a negative impact on the promotion of cooperative behavior. To overcome this issue, the authors suggest assigning higher weight to the collective knowledge for strategy updating process.

The promotion of cooperation in INPPD is also presented in [24]. The authors investigate the application of co-evolutionary training techniques based on PSO to evolve the cooperative behavior of INPPD games. The simulation results show that three factors affect the promotion of cooperation in INPPD populations. These factors are mainly related to the length of the history record which players can access during the game, the ratio of the cost the players are paying against the benefits they can get for a specific action, and finally the size of the group in which the players are interacting.

Similarly, the authors in [25] discuss the power of PSO in evolving the cooperation behavior among INPPD players. Beside the evolution of cooperation, the authors addressed the issue of evolutionary stability in noisy environments. Experimental results show that PSO evolves the cooperative behavior between INPPD players. It is also noted that players with strong social mentality with other players choose higher levels of cooperation with no impact of noise on the overall level of cooperation.

The research presented in [26] presents a PSO-based model for evolving the cooperation behavior among selfish INPPD individuals. The model aims to simulate the behavior of swarms' particles over the INPPD players to track the players with the highest payoff attained within a local topological neighborhood. However, the simulation results reveal that PSO was able to significantly increase the level of cooperation in the population in such environments that strongly favor the proliferation of defection.

As a conclusion, we found that none of the existing models have considered the ultimate utilization of best players' experiences in the game. Existing models give equal consideration for all players in the game while changing only the communication topology. Adopting specific

Journal of Theoretical and Applied Information Technology
15th March 2019. Vol.97. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

topology in PSO results in assigning higher weights for only the best players in the local topological neighborhood. This issue motivates us to look beyond the existing communication topologies and construct an evolutionary model which focusses on the capabilities of the best players with social awareness to evolve the cooperative behavior among INPPD players.

5. RELATED WORKS

In this section we introduce our evolutionary model which is based on PSO. The pivotal components forming the base of our model are: the use of PSO as evolutionary algorithm, the adoption of and alternative neighborhood topological structure to ensure proper sharing of experiences between the players, and the utilization of knowledge base to support the players' decision making process.

5.1 Neighborhood Topological Structure

Specifying the neighborhood topological structure is essential for the players before starting communication. It is one of the most important aspects that affect the performance of PSO. The topology refers to the form on which each particle is connected with its neighborhoods. Practically, too loose a topology forces the particles to spend too much effort in low quality regions of the space, which leaves promising regions exploited by just a little amount of elements, while, a highly connected topology could make the particles collapse too quickly, making the system easily trapped in local optimum. Hence, a good topology should properly support the exploration of promising regions in the search space, and allow the existence of several search spots with easy exchange of information among them.

In this model we present an alternative communication topology which allows the best players to propagate their experience among all other players. Our alternative topology divides the particles into sub-swarms. Particles in each sub-swarm are communicating through Von-Neumann topology, as well as the communicating with the best particles from the neighboring sub-swarms. Our topology considers Von-Neumann topology for supporting intra-communication between sub-swarm as illustrated in Fig. 2. This design prevents particles from wasting their efforts in low quality regions as in too loose topology, and at the same time, prevents particles from becoming trapped in local optimum as in highly connected topologies. Practically, each particle in PSO chooses its moves according to the behavior of its own sub-swarm, as well as the behavior its neighboring sub-swarms.

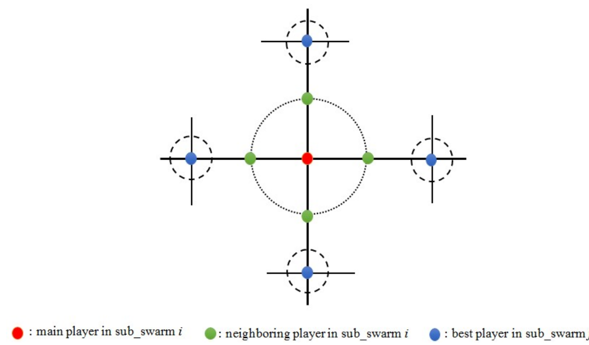

Figure 2: Alternative communication topology between PSO particles (INPPD players)

5.2 Players' Knowledge Base

The model facilitates the players to access the history repository of best players' behavior. This repository is designed to be evolving as the players making decisions through the iterations of INPPD games. Therefore, this repository is defined as knowledge base in our model. As players' decisions represent our data, the knowledge base aims to generate meaningful knowledge structures that are hidden in these data. This is to aid the decision making process of the players.

The knowing process is taken place when a particular player has some information that might help enhancing the performance (payoffs) of the player itself or the performance of other players. These payoffs are mainly calculated based on the learned knowledge during the game. A knowledge base of the game is constructed in the main memory, and the team strategy is mostly made using a knowledge base. However, the player is said to have knowledge if it knows, with a probability, what is the outcome of the action it may perform, or the actions performed by other player.

Our knowledge base consists of two main rational entities. The first entity covers the sub-swarm activities, while the second entity covers the swarm (whole population) activities. The sub-swarm entities are designed to store players' moves and strategies adopted within each sub-swarm. Each sub-swarm has a unique identifier in each entity which distinguishes it from other sub-swarms. These entities are also tracking the best neighbors (with respect to our neighborhood communication topology) of each sub-swarm.

The swarm entities are designed to track the best strategies achieved by the population, finding the global best players and the current positions of all

Journal of Theoretical and Applied Information Technology
15th March 2019. Vol.97. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

players to assist PSO tracking of the best players in the population and support the global best orientation. Both sub-swarm and swarm entities assist the model to enhance the cooperation ratio between players and in predicting the opponents' behaviors, which allows the players to adopt the countermeasure strategy that can defeat the opponents' strategy. However, the knowledge base in our model acts as an analytical tool of the players' strategies and for supporting our PSO to find the best solution among *n* possible solutions.

5.3 PSO-based Evolutionary Model

Each particle in PSO aims to find the best position while they move through the problem space. These particles are randomly initialized by a random velocity and position at the beginning of the searching. Each particle adjusts its position in a direction toward its own personal best position and the neighborhood best position. In this model we represent each group of players as a sub-swarm of particles.

Finding neighborhood's best particle depends on the communication topology being used. For that reason, the knowledge base consists of all information that helps the model finding the best neighbors based on the adopted neighborhood structure. Our topological neighborhood structure (as described in section 5.1) aims to identify the best *pn* players among the participating players. This is to provide the population with extra connections with the best players in the population and ensure diversity and experience sharing between INPPD participants.

INPPD is carried out by a number of agents in a specific number of generations, games, and moves. The generation is composed of multiple games, and each game is operated by multiple moves. When the number of agents, generations, games and moves are increased, INPPD will tend to be more complex. Hence, our evolutionary model aims to effectively handle the increasing number of players in INPPD games.

To enhance the cooperation ratio and generate competitive strategies of INPPD game, PSO particle flies through the game space to find the best possible position among all possible positions. The initial position (X_i) and velocity (V_i) of each particle are randomly chosen at the beginning of the search. During the search process, PSO particles change their positions and velocities (referred by behavior) based on the behavior of their close neighbors and the particle's behavior itself. PSO particles are equipped with a memory for its best personal position (P_{best}) and a memory for its best neighborhood position (N_{best}). Each particle aims to change its own position and velocity under the guidance of its own experience (with respect to its P_{best}) and its neighbors' experience (with respect to the N_{best} value) to reach the best possible area (global optimum). In addition, we consider one extra parameter (Gn_{best}) which identifies a set of the best particle position in the population. These three parameters are continuously updated as our knowledge base continues to evolve.

In early generations of PSO, particles have insufficient level of knowledge about their neighbors' behaviors. Therefore, the expected payoff of most particles will not exceed t, such that (where Mt is the maximum possible payoff value):

$$t < M_t \quad (1)$$

As the generations are passing through, a particle's knowledge will be reinforced by its own experience and its neighbor's behavior. The reinforcement of a particle's knowledge results in evolving the particles strategies through the generations. However, the evolved strategies can achieve higher payoff (et) values such that (where e is a constant):

$$et < M_t \quad (2)$$

The evolvement of INPPD player's strategy is affected by its corresponding PSO particle flying over the searching space. At the end of generation, each PSO particle compares its best personal position with the other particles in its sub-swarm and the swarm. Before starting a new generation, each particle should update its own position and velocity according to the behavior of the best particles (sub-swarm and swarm levels).

The position X_i of a given particle is updated with respect to its velocity V_i according to:

$$X_i(t+1) = X_i(t) + V_i(t+1) \quad (3)$$

where the velocity of the particle is calculated using the formula in Eq. (4):

$$V_i(t+1)=wV_i(t)+c_1r_1(t)(y_i(t)-x_i(t))+c_2r_2(t)(\bar{y}_i(t)-x_i(t)) \quad (4)$$

such that the parameters r_1, r_2 represent two random numbers such that $r_1,r_2 \sim U(0,1)^n$, y_i denotes the personal best position of particle i, and $\bar{y}_i$ denotes the neighborhood best position with respect to our communication topology. This parameter is computed as shown in Eq. (5):

$$\bar{y}_i(t+1) \in \{N_i | f(\bar{y}_i(t+1)) = max\{f(x), \forall x \in N_i\} \quad (5)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where the neighborhood N_i of neighborhood size l is defined in Eq. (6) as follows:

$$N_i = \{y_{i-1}(t), \dots, y_{i-1}(t), y_i(t), y_{i+1}(t), \dots, y_{i-1}(t)\} \quad (6)$$

The two coefficients c_1, c_2 are two time-varying acceleration coefficients, which are developed by Ratnaweera to modify the local and the global search ability and increase the diversity [27]. The mechanism of these two factors is to linearly reduce $c1$ and increase c_2 with time. This mechanism has a significant impact on enhancing the global search ability of PSO at the early stages of the search, and at the same time, improving the local search ability at the end of the search. However, the values assigned to the acceleration coefficients should balance between global and local search ability of PSO. If the difference between c_1 and c_2 is larger than 1, the convergence accuracy stability of PSO becomes poor. Generally, the acceleration coefficient c_1 governs the individual experience of each particle, while the coefficient c_2 governs the social communication between particles. In our model we set c_1 and c_2 to 2.0 and 2.5 respectively.

The parameter w in Eq. (4) is the inertia weight parameter that is first introduced by Shi and Eberhart [28]. This parameter is created to balance the global exploration and the local exploitation of PSO. Inertia weight w is a parameter within the range [0, 1] and is often decreased over time to control the impact of the previous history of velocities on the current velocity. The importance of this parameter comes from its ability in influencing the trade-off between global and local exploration capabilities of the flying particles. The experiments conducted by Shi and Eberhart showed that a larger inertia weight facilitates global exploration while a smaller inertia weight facilitates local exploration. Thus, setting the suitable inertia weight is responsible for providing a balance between global and local exploration abilities, resulting in a less number of iterations to find the optimum. In our model, we set the parameter w to the initial weight 1.0 (w=1.0) as recommended in [29]. The initial value of w is decreased from 1.0 to 0.1 as the particles fly over the PSO search space. Starting from 1.0 and then decrementing the inertia weight toward 0.1 is important to promote exploration in early optimization stages, and to eliminate oscillatory behaviors in later stages. Note that the lower bound is set to 0.1 to prevent the previous velocity term from disappearing.

As INPPD players represent PSO particles, each particle starts changing its own behavior based on the quality (specified through the payoff) of its own previous behavior and the behavior of its neighboring particles (according to our alternative communication topology). This technique allows players to change their low-quality behavior by adopting the behavior of neighboring players. After completing m generations, the players start to move toward promising regions which includes the best possible game strategy. At the end of each generation, PSO evaluates the fitness of each agent. That fitness specifies the best possible agent in the population. Note that the best agent refers to its strategy which achieved the best fitness compared to other agents. However, PSO encourages other agents to follow the best agent by adopting its strategy.

Agents communicate with each other's in several communication topologies. Nevertheless, our population is divided into sub-swarms (group). Each sub-swarm has a fixed number of agents and it allows its agents to communicate thoroughly. When PSO is activated, each sub-swarm will have its own local best agent, and the whole population will, accordingly, have one single global best agent.

Many problems may arise when the number of players is increased. These problems include the complexity of representing players' strategies and the slowness in convergence towards the optimal solution due to the insufficiency of the communication level provided by traditional topologies. Therefore, in our model we presented the knowledge base component to assist PSO in making better decisions and providing INPPD agents with clearer vision on the performance of wider range of agents in the population.

6. PERFORMANCE ANALYSIS

The evolutionary model presented in this paper focuses on enhancing the cooperation ratio among INPPD rational players and enables players to evolve their strategies to survive even when they play against the most defective players. In this section we evaluate our model in term of cooperative ratios as well as the capability of evolving competing strategies against well-known strategies.

To standardize the evaluation process, we initialized the INPPD game as shown in Table 3. These parameters are tuned to measure the performance of the model from different angles. Note that the values of this table are used in all evaluation tests carried out in this section.

Table 3: INPPD Initialization

Parameter	Initialized Value
no. of generations	2000
no. of games	50
no. of moves	50
no. of players in population (*n*)	30-100
no. of simulations	20

The evaluation tests carried out in this section is divided into two categories. The first category is concerned in analyzing the performance of INPPD players (as individual player, as sub-swarm and as population) using our evolutionary model. The second category involves comparative testing against benchmark strategies. Note that our evolutionary model is denoted by PSO-*evo* while traditional PSO model is denoted by PSO for easier comparisons.

Based on the parameters setting shown in Table 3, we start by testing PSO-*evo* on INPPD by tuning the values of the model parameters. This part of analysis aims to show the efficiency of our model in increasing the number of cooperators in INPPD games. Fig. 3 shows how PSO-*evo* could generally enhance the performance of INPPD players.

Obviously, PSO-*evo* has achieved better results when tested on both population sizes. We noticed that PSO achieves better results on early generations as the players are allowed to play randomly at the early stages of INPPD game. Nevertheless, the performance is changed dramatically as our communication topology along with the evolving of the knowledge base are operating during the new generations.

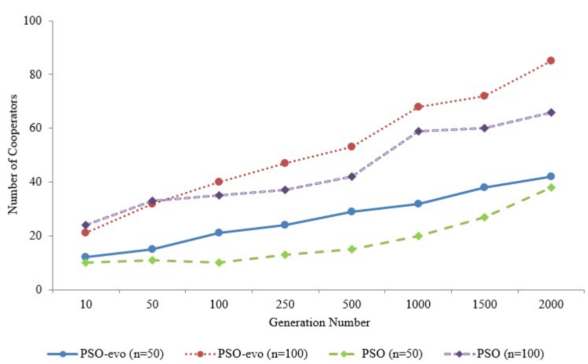

Fig.2: Performance of PSO-evo and PSO on INPPD of size 50 and 100

The performance of the individual player has an impact on the performance of its neighbors in the sub-swarm, particularly, and on the whole population, generally. Hence, we examine the impact of our model in enhancing the performance of the sub-swarms and the population as a whole. The tests aim to measure the ability of PSO-*evo* in leading the PSO particles to the optimal possible positions. A population of 30, 50 and 100 particles are examined and analyzed. In this part of the evaluation we choose the best sub-swarm and compare its behavior (with respect to its payoff) with the average behavior of the population in 2000 generations. Fig. 4 shows that the best sub-swarm could achieve average payoffs which are almost close to the average payoffs achieved by the whole population. This indicates that particles within the whole population are sharing their experiences effectively.

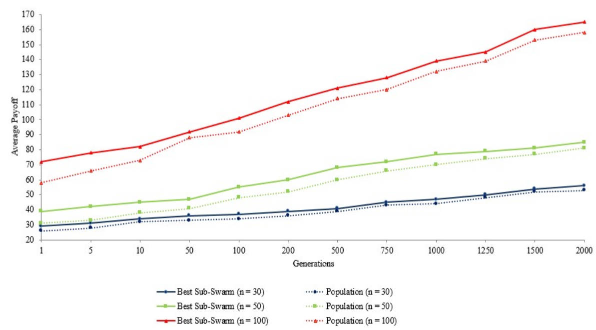

Fig.4: Correlation between INPPD sub-swarm's performance and population's performance

It is obvious that the difference between the average payoff achieved by the best sub-swarm and the average payoff achieved by the whole population (on different population sizes) is negligible. This is due to the efficiency of our alternative neighborhood communication topology which facilitates effective communication between the participated sub-swarms.

In [30], the experiments showed that the occurrence of the following two conditions during a specific generation indicates that a cooperative behavior is approaching within the population:

- **Condition 1**: the total number of cooperation actions made by a particular player with the largest payoff is ten times greater than the population size.
- **Condition 2**: the total number of cooperative actions made by the whole population is ten times greater than the population sizes.

Satisfying these two conditions indicate that the population favors cooperation. Based on the initial settings stated in Table 3, our simulations are carried out on a population of size 30. Each generation is composed of 50 games and each game consists of 50 moves. Therefore, the total number of actions made by each player in any given generation is 2500 actions.

We have tested our results against the conditions 1 and 2 and we found that our population favors the cooperative behavior rather than the defective behavior. Fig. 5 shows that the best player among the 30 players has played more than 300 cooperative actions (where 300 is ten times the population size) in less than 1000 generation. Moreover, the number of cooperative actions played by the best player is found to be 30 times greater than the population size in the last generation.

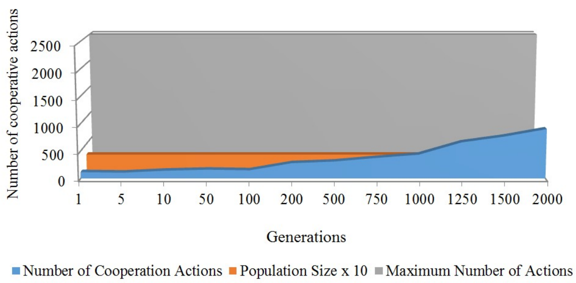

Fig.5: Measuring the behavior of the population (n = 30) in term of the number of cooperative actions made by the best player

Increasing the population size played a pivotal role in decreasing the number of cooperative actions made by the best player. We have tested PSO-*evo* on two more population sizes of 50 and 100. When the population size is increased to (n =100), PSO-*evo* satisfied 70% of the first condition where the best player could make a number of cooperative actions which is approximately 7 times greater than the population size. Fig. 6 illustrates the impact of population size on the number of cooperative actions made by the best player of the corresponding populations.

On the other hand, we have tested our model against the second condition and the results showed that PSO-*evo* has satisfied this condition efficiently with populations of sizes (n = 30) and (n = 50) as shown by Fig. 7. With larger populations, PSO-*eco* showed less efficiency in satisfying this condition. A population with 100 players could achieve an average number of cooperative actions that is about 6 times the population size.

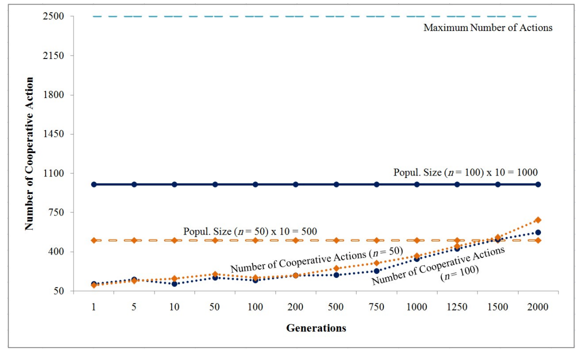

Fig.6: Measuring the behavior of the populations (n = 50, n = 100) in term of the number of cooperative actions made by the best player

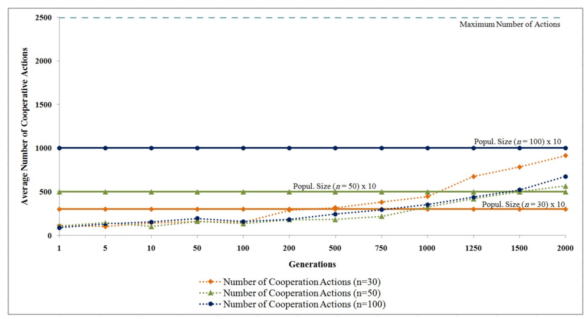

Fig.7: Measuring the behavior of the populations (n = 30, n = 50 and n = 100) in term of the average number of cooperative actions made by the population

However, our model is still able to survive and drag players towards cooperative behaviors even in large populations. In the following test, we examine our model against well-known benchmark strategies which are extensively used by many important researches dealing with Prisoner's Dilemma. The results aim to show the total payoff achieved by each strategy when played against other benchmark strategies as well as the efficiency of each strategy compared to the standard benchmark measurements. Based on the simulation parameters given in Table 3, we examine our generated strategies against 10 well-known benchmark strategies. These strategies are described in Table 4.

Testing our strategies against benchmarking strategies is essential to show the capability of our model in generating efficient strategies that can defeat selfish strategies (favoring defections). Research showed that these strategies proved to be hard-to-defeat in 2IPD games. Therefore, we examined the strength of these benchmark strategies in surviving in INPPD game.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

This is to measure the efficiency of the competent strategies with respect to their average scores. At the end of each INPPD generation, the game will produce n strategies (one strategy for each player). The strategy of each player is selected based on the highest payoff that could be achieved at that particular generation. The selected n strategies in generation i is made to play against 10 other opponent strategies (benchmark strategies). However, all strategies (n + 10) are moved to the competition pool where they play against each other accordingly.

Calculating the playoff of each strategy depends on the same INPPD payoff matrix presented in Table 1. For instance, given an INPPD of 10 players, our model will generate 10 strategies at the end of each generation. The strategy of player i in generation r is denoted by $SP_{r,i}$, where $i \in \{1,2,3,\ldots,10\}$ and $r \in \{1,2,3,\ldots,m\}$. These strategies play against the other 10 benchmark strategies. Based on the adopted payoff matrix (**F**), each strategy will score a specific payoff on every action it takes.

The payoff function (**F**) calculates the number of cooperated players ($\mathbf{C}_r$) among the other n-1 players. This number determines the payoff amount that each player should obtain based on the payoff matrix. For instance, Table 5 shows a list of three moves generated by the participated players. Note that M_i denotes the current move number, 'M_i Score' denotes the payoff of M_i and the 'Total Score' denotes the total payoff that each strategy could achieve during each tournament.

Table 4: Benchmark Strategies

Strategy Code	Strategy	Description
AC	*Always Cooperate*	Cooperates on every move
AD	*Always Defect*	Defects on every move
TFT	*Tit-for-Tat*	Cooperates on the first move, and then copies the opponent's last move
STFT	*Suspicious Tit-for-Tat*	Same as TFT except that it defects in the first move
TFTT	*Tit-for-two-Tat*	Cooperates on the first move and defects only when the opponent defects two times
HTFT	*Hard Tit-for-Tat*	Cooperates on the first move and defects only if the opponent has defects on any of the previous three moves
PAV	*Pavlov*	Cooperates on the first move and defects only if both players did not agree on the previous move
SPT	*Spiteful*	Cooperates until the opponent defect, and then always defect.
SMJ	*Soft Majority*	Start cooperating, and cooperates as long as the number of times the opponent has cooperated is greater than or equal to the number of times it has defected, else it defects.
HMJ	*Hard Majority*	Defects on the first move, and defects if the number of defections of the opponent is greater than or equal to the number of times it has cooperated, else cooperates

The example shown in Table 5 indicates that both of $\mathbf{SP}_{1,8}$ and **HMJ** strategies could achieve the highest payoff (total score of **65**) among the other strategies. The score of each move is calculated as follows: for the move $\mathbf{M_1}$ of each strategy, the function **F** found that the total number of cooperated moves was $\mathbf{C}_r$ = 12. That means, each player with **C** move is awarded a payoff of a value 2(12-1)=22, while the rest of players with **D** move are awarded 2(8-1)+1=15. The same procedure is applied on all subsequent moves.

Table 5 Illustrative example on calculating the payoff scores in IPD of 20 players

Strategy	M_1	M_1 Score	M_2	M_2 Score	M_3	M_3 Score	Total Score
$\mathbf{SP}_{1,1}$	*C*	22	*D*	23	*D*	17	62
$\mathbf{SP}_{1,2}$	*C*	22	*C*	14	*D*	17	53
$\mathbf{SP}_{1,3}$	*D*	15	*D*	23	*D*	17	55
$\mathbf{SP}_{1,4}$	*D*	15	*D*	23	*D*	17	55
$\mathbf{SP}_{1,5}$	*C*	22	*D*	23	*D*	17	62
$\mathbf{SP}_{1,6}$	*D*	15	*D*	23	*C*	20	58
$\mathbf{SP}_{1,7}$	*C*	22	*C*	14	*C*	20	56
$\mathbf{SP}_{1,8}$	*C*	22	*D*	23	*C*	20	**65**
$\mathbf{SP}_{1,9}$	*D*	15	*D*	23	*C*	20	58

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$SP_{1,10}$	*C*	22	*D*	23	*D*	17	62
AC	*C*	22	*C*	14	*C*	20	56
AD	*D*	15	*D*	23	*D*	17	55
TFT	*D*	15	*D*	23	*C*	20	58
STFT	*C*	22	*C*	14	*C*	20	56
TFTT	*D*	15	*C*	14	*C*	20	49
HTFT	*D*	15	*C*	14	*C*	20	49
PAV	*C*	22	*D*	23	*D*	17	62
SPT	*C*	22	*C*	14	*C*	20	56
SMJ	*C*	22	*C*	14	*D*	17	53
HMJ	*C*	22	*D*	23	*C*	20	**65**

In order to examine the strength of our strategies against the benchmark strategies, a complete INPPD game of 40 players is initiated. Among the participated players, we have 10 players who are assigned to play the selected ten benchmark strategies as fixed strategies throughout the game. The rest of the players are changing their behavior according to our PSO-evo model's suggestions. The results presented in Table 6 (ordered from highest to lowest payoff scores) show the average scores of PSO-*evo* strategies when played against the selected benchmark strategies in 2000 generations. The average payoff score is calculated by summing the payoff scores achieved by each strategy in each generation, and then divides it over the total number of generations. The maximum payoff that a strategy can achieve in each generation is 2(40-1)+1=79. Therefore, after completing 2000 generations, the maximum payoff that a strategy can achieve should not exceed 158000.

Table 6 Average scores of PSO-evo strategies against benchmark strategies (40 players)

Rank	Strategy	Average Score	Rank	Strategy	Average Score
1	SP_{25}	72	21	SP_{2}	44
2	SP_{14}	68	22	**SMJ**	39
3	**TFT**	67	23	SP_{21}	37
4	SP_{1}	65	24	SP_{5}	36
5	SP_{8}	65	25	**SPT**	35
6	**HTFT**	62	26	SP_{24}	35
7	SP_{13}	60	27	SP_{3}	34
8	**TFTT**	58	28	SP_{20}	30
9	SP_{22}	57	29	**AC**	28
10	SP_{10}	57	30	SP_{30}	25
11	SP_{16}	57	31	SP_{4}	25
12	**STFT**	56	32	SP_{15}	25
13	SP_{28}	55	33	SP_{7}	24
14	SP_{19}	54	34	SP_{17}	24
15	**HMJ**	53	35	SP_{23}	22
16	SP_{11}	53	36	**AD**	19
17	SP_{27}	52	37	SP_{9}	18
18	SP_{6}	52	38	SP_{12}	18
19	**PAV**	50	39	SP_{26}	17
20	SP_{29}	45	40	SP_{18}	15

Each strategy has an efficiency value which reflects its performance with respect to the maximum payoff value that can be achieved. For instance, the efficiency of SP_{25} is found to be (72 ÷ 79 = 91.1%). Fig. 8 reflects the performance efficiency of PSO-*evo*'s strategies against benchmark strategies in the first tournament.

In the second tournament, we increased the number of participated strategies to 60. The experiment results showed that PSO-*evo* strategies could also defeat the benchmarking strategies. From the other side, we noticed that the performance of benchmark strategies was degraded, resulting in a lower ranking for these strategies as obviously seen in Table 7.

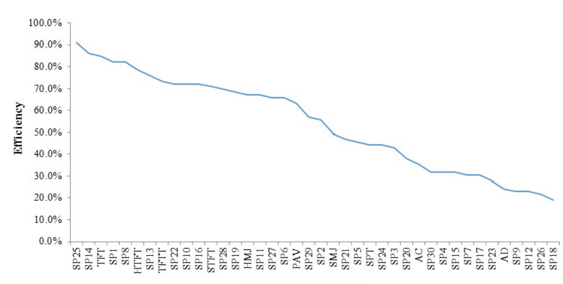

Fig.8: The Performances Efficiency of PSO-evo Strategies against Benchmark Strategies (Tournament-1 of 40 players)

Journal of Theoretical and Applied Information Technology
15th March 2019. Vol.97. No 5

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 7 Average Scores of PSO-evo Strategies against Benchmark Strategies (60 players)

Rank	Strategy	Average Score	Rank	Strategy	Average Score
1	SP_{34}	70	31	SP_{35}	48
2	SP_{16}	70	32	SP_{2}	47
3	SP_{22}	70	33	SP_{10}	45
4	SP_{9}	69	34	SP_{23}	44
5	SP_{12}	68	35	**PAV**	43
6	SP_{27}	68	36	SP_{37}	38
7	SP_{43}	68	37	SP_{3}	37
8	SP_{14}	67	38	SP_{17}	35
9	**TFT**	67	39	SP_{29}	35
10	SP_{11}	67	40	SP_{26}	35
11	SP_{32}	62	41	SP_{38}	31
12	SP_{36}	61	42	**SMJ**	29
13	**HTFT**	61	43	SP_{20}	28
14	SP_{6}	61	44	SP_{48}	28
15	**HMJ**	60	45	SP_{19}	25
16	SP_{8}	57	46	**AC**	22
17	SP_{31}	56	47	SP_{46}	22
18	SP_{44}	56	48	SP_{1}	21
19	SP_{15}	55	49	SP_{30}	21
20	**TFTT**	54	50	SP_{25}	20
21	SP_{33}	54	51	**AD**	19
22	SP_{45}	54	52	SP_{7}	17
23	SP_{4}	53	53	SP_{49}	17
24	SP_{18}	53	54	SP_{41}	17
25	**SPT**	51	55	SP_{47}	17
26	SP_{5}	50	56	SP_{40}	16
27	SP_{50}	49	57	SP_{39}	14
28	SP_{24}	49	58	SP_{28}	14
29	**STFT**	48	59	SP_{21}	14
30	SP_{13}	48	60	SP_{42}	11

In the third tournament, the number of participated strategies was increased to 100 strategies. The experiment results, presented in Table 8, showed that PSO-*evo* strategies could defeat the benchmarking strategies. We also noticed that only **TFT** strategy could survive in the top 10 strategies. As a result, the efficiency of benchmark strategies is found to be degraded as we increase the number of participated strategies in the competition pool.

Table 8 Average scores (Avg.) of PSO-evo strategies against benchmark strategies (Str.) (100 players)

Str.	Avg.	Str.	Avg.	Str.	Avg.	Str.	Avg.
SP_{12}	74	SP_{51}	55	SP_{87}	37	SP_{67}	15
SP_{59}	74	SP_{54}	55	SP_{10}	35	SP_{34}	14
SP_{36}	72	SP_{22}	54	SP_{39}	32	SP_{11}	14
SP_{8}	72	SP_{26}	53	SP_{42}	32	**PAV**	14
SP_{44}	72	SP_{5}	53	SP_{57}	29	SP_{43}	13
TFT	69	SP_{85}	53	SP_{75}	28	SP_{13}	13
SP_{79}	69	SP_{16}	53	SP_{62}	25	SP_{45}	12
SP_{88}	69	SP_{56}	52	SP_{63}	25	SP_{21}	12
SP_{74}	68	SP_{38}	51	SP_{32}	25	SP_{25}	12
SP_{19}	68	SP_{50}	50	SP_{84}	25	SP_{49}	11
SP_{64}	68	SP_{20}	50	**SMJ**	25		
SP_{86}	66	**HTFT**	50	SP_{29}	24		
SP_{6}	66	SP_{55}	48	SP_{90}	24		
SP_{77}	65	SP_{33}	48	SP_{53}	24		
SP_{7}	64	SP_{58}	47	SP_{48}	23		
SP_{1}	64	**AC**	46	SP_{27}	23		
SP_{40}	64	SP_{71}	45	SP_{15}	23		
SP_{68}	64	SP_{17}	45	SP_{73}	22		
SP_{82}	63	SP_{83}	45	SP_{46}	21		
SP_{69}	63	**HMJ**	40	**TFTT**	20		
SP_{3}	61	SP_{4}	40	SP_{9}	20		
SP_{80}	61	SP_{47}	40	SP_{81}	20		
SP_{66}	61	SP_{60}	39	SP_{23}	20		
SP_{89}	60	SP_{41}	39	SP_{52}	19		
SP_{61}	58	SP_{24}	39	**AD**	19		
SP_{31}	58	SP_{72}	38	SP_{18}	18		
SP_{2}	57	SP_{14}	38	SP_{76}	18		
SP_{28}	57	**SPT**	38	SP_{70}	18		
STFT	56	SP_{65}	38	SP_{37}	17		
SP_{30}	56	SP_{78}	38	SP_{35}	16		

After analyzing the performance of PSO-*evo* strategies against the benchmark strategies, we found that the performance of benchmark strategies is inversely proportional to the number of the participated strategies in the game. As we increase the number of players, the performance of the benchmark strategies is degraded. Fig. 9 illustrates

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the impact of increasing the number of players on reducing the number of benchmark strategies in the winning top 10 strategies.

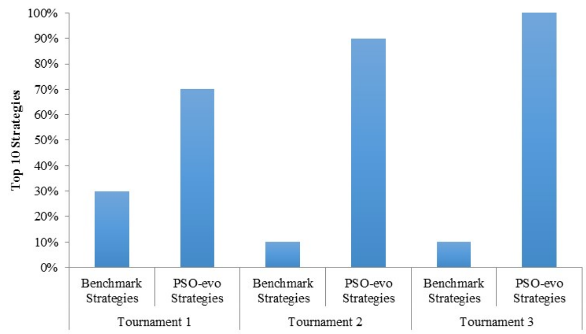

Fig.9: List of top 10 Strategies in the First, Second and Third Tournaments

In order to confirm the efficiency of our model in generating competent strategies, we conduct a comparative study against other existing models. These models claimed to generate competitive strategies which can defeat benchmark strategies. Among these models we choose the models published in [31]-[33] as we relatively share similar platforms. The strategies generated by [31]-[33] are denoted by *Freud*, *Strategy-MO* and *Gradual*, respectively.

In this test, we measure the average payoff of each model against five benchmark strategies, including: TFT, TFTT, PAVLOV, AC and AD. The reason for choosing only five benchmark strategies is that the chosen models choose these particular benchmark strategies in their tests.

The results presented in Table 9 shows that PSO_*evo* strategies outperforms the benchmark strategies as well as the other three models as it could achieve the highest approximate average payoff among the other strategies. These results indicate that PSO-*evo* strategies could generate complex strategies to survive in complex environments.

Table 9 Average Payoff of the Competent Models against Benchmark Strategies

	Avg. Payoff	TFT	TFTT	PAVLOV	AC	AD
Freud	559	540	547	462	392	427
Strategy-Mo	448	391	370	334	337	396
Gradual	334	314	N/A	289	255	244
PSO-*evo*	720	670	580	500	280	190

From the other perspective, one can easily note that TFT strategy has higher performance than the other benchmark strategies. Fig. 10 shows a performance comparison between the competent 4 models against the selected five benchmark strategies. The results show that the selected benchmark strategies have lower performance on PSO-*evo* strategies when compared to Freud, Strategy-MO and Gradual strategies.

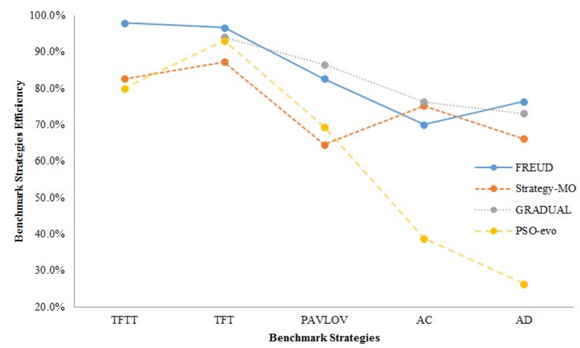

Fig.10: Performance of the benchmark strategies against the competent models

The experiments results show that our model has successfully evolved the cooperative behaviour among the players to achieve the best possible outcomes. Our model is tested against well-known models and the performance results shows that the cooperative behaviour of the players adopting our model, could achieve better scores compared to other models.

7. CONCLUSION

In this paper we have investigated the evolution of cooperative behavior in INPPD. For the purpose of enhancing the cooperation rate in large INPPD populations, an evolutionary model was presented using the PSO. The model was able strengthen the communication between INPPD players at different levels, which is essential for establishing proper collaboration. The utilization of the evolving knowledge base played a primary role in assisting INPPD players in predicting their opponents' behavior though game's generations.

The results showed that our evolutionary model was able to enhance the cooperation rate in INPPD games of large populations. The results also showed that our model could help INPPD players to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

strongly survive against defectors and those who are equipped with static benchmark strategies.

REFRENCES:

[1] K. Sigmund and M. A. Nowak. "Evolutionary game theory". *Current Biology*, 9(14):R503–505, 1999.

[2] J. M. Houston, J. Kinnie, B. Lupo, C. Terry, and S. S. Ho. "Competitiveness and conflict behavior in simulation of a social dilemma". *Psychological Reports*, 86:1219–1225, 2000.

[3] D. Roy. "Learning and the theory of games". *Journal of Theoretical Biology*, 204:409–414, 2000.

[4] M. Hemesath. "Cooperate or defect? Russian and American students in a prisoner's dilemma". *Comparative Economics Studies*, 176:83–93, 1994.

[5] X. Wang, L. Zhang, X. Du and Y. Sun, "Effects of Punishment in a Structure Population Playing the Social Dilemma Game by Using Discrete PSO Algorithm", *in Proc. IEEE International Conference on Smart Cities* (Guadalajara, Mexico, 2015) , pp. 407 – 414.

[6] W. Zeng, M. Li and F. Chen, "Cooperation in the evolutionary iterated prisoner's dilemma game with risk attitude adaptation", *Applied Soft Computing* 44:238-254. 2016.

[7] K. Sorgi and M. Wout, "The influence of cooperation and defection on social decision making in depression: A study of the iterated Prisoner's Dilemma Game", *Psychiatry Research* 246:512-519. 2016.

[8] X. Yao and P. Darwen, "An experimental study of N-person iterated prisoner's dilemma games". *Progress in Evolutionary Computation* 956:90-108. 1995.

[9] J. Kennedy and R. Mendes, "Population structure and particle swarm performance", *in Proc. IEEE Congress on Evolutionary Computation*, (USA, Hawaii, 2002), pp. 1671–1676.

[10] E. Peer, F. Bergh and A. Engelbrecht, "Using Neighborhoods with the Guaranteed Convergence PSO", *in Proc. IEEE Swarm Intelligence Symposium* (Indianapolis, USA, 2003), pp. 235–242.

[11] I. Guerberoff, L. Bo and J. Sichman, "Strategy representation complexity in an evolutionary n-Players Prisoner's Dilemma model". *XVI Seminário de Computação*. FURB: Universidade Regional de Blumenau. 2007.

[12] Yudong Zhang, Shuihua Wang, and Genlin Ji, "A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications" *Mathematical Problems in Engineering*, vol. 2015, Article ID 931256, 38 pages, 2015.

[13] C. Wang, K. Liu, "A Novel Particle Swarm Optimization Algorithm for Global Optimization", *Computational Intelligence and Neuroscience*, vol. 2016, ID 9482073, 9 pages. 2016.

[14] Mukhopadhyay, S., & Banerjee, S. "Cooperating Swarms: A Paradigm for Collective Intelligence and its Application in Finance". *International Journal of Computer Applications*, 6 (10), 0975 – 8887. 2010.

[15] Evers, G. "An Automatic Regrouping Mechanism to Deal with Stagnation in Particle Swarm Optimization", M.Sc. dissertation, The University of Texas-Pan American Edinburg, TX, 2009.

[16] T. Sudo, K. Goto, Y. Nojima and H. Ishibuchi, "Effects of ensemble action selection with different usage of player's memory resource on the evolution of cooperative strategies for iterated prisoner's dilemma game", *in Proc. IEEE Congress on Evolutionary Computation* (Sendai, Japan, 2015), pp. 1505 – 1512.

[17] F. Kenter and E. Meigs, "Iterated prisoner's dilemma with extortionate zero-determinant strategies and random-memory opponents", *in Proc. IEEE Congress on Evolutionary Computation* (Vancouver, Canada, 2016), pp. 3499 – 3506.

[18] D. Kasthurirathna, M. Piraveenan and S. Uddin, "Evolutionary stable strategies in networked games: the influence of topology", *Journal of Artificial Intelligence and Soft Computing Research* 5(2):83–95. 2015.

[19] D. Ashlock and E. Kim, "The impact of elite fraction and population size on evolved iterated prisoner's dilemma agents", *in Proc. IEEE Congress on Evolutionary Computation* (Vancouver, Canada, 2016), pp. 364 – 371.

[20] L. Barlow and D. Ashlock. "The impact of connection topology and agent size on cooperation in the iterated prisoner's dilemma", *in Proc. IEEE Conference on Computational Intelligence in Games* (Niagara Falls, Canada, 2013), pp. 299-306.

[21] L. Menglin, "Graph topology and the evolution of cooperation", PhD Thesis, NUI Galway, 2015.

[22] X. Wang, L. Zhang, X. Du and Y. Sun, "Effects of Punishment in a Structure Population Playing

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the Social Dilemma Game by Using Discrete PSO Algorithm", *in Proc. IEEE International Conference on Smart Cities* (Guadalajara, Mexico, 2015) , pp. 407 – 414.

[23] Duan, H., Sun, C., "Swarm intelligence inspired shills and the evolution of cooperation", *Scientific Reports* 4, Article number: 5210. 2014.

[24] Xiaoyang, W., Lei, Z., Xiaorong, D., "Using Discrete PSO Algorithm to Evolve Multi-player Games on Spatial Structure Environment", *Advances in Swarm and Computational Intelligence*, vol. 9141 of the series Lecture Notes in Computer Science pp 219-228. 2015.

[25] Xiaoyang, W., Yang, Y., Huiyou, C., Yibin, L., "Using Particle Swarm Optimization To Evolve Cooperation In Multiple Choices Iterated Prisoner's Dilemma Game", *Int. J. Comp. Intel. Appl*. 12, 1350013. 2013.

[26] Zhang, J., Zhang, C., Chu, T., Perc, M., "Resolution of the Stochastic Strategy Spatial Prisoner's Dilemma by Means of Particle Swarm Optimization", *PloS ONE* 6(7): e21787. 2011.

[27] Sun, P., Sun, H., Feng, W., Zhao, Q., & Zhao, H. "A Study of Acceleration Coefficients in Particle Swarm Optimization Algorithm Based on CPSO". 2010 *2nd International Conference on Information Engineering and Computer Science* (ICIECS, 2010), (pp. 1-4).

[28] Shi, Y., & Eberhart, R. C. "Parameter Selection in Particle Swarm Optimization". *Evolutionary Programming* VII (pp. 591–600). Berlin, Germany: Springer-Verlag. 1997.

[29] Saharan, L., & Mittal, M. "Comparison of Various PSO Variants with Different Mathematical Functions". *International Journal on Theoretical and Applied Research in Mechanical Engineering*, 2 (3), 135-142. 2013.

[30] Franken, N., & Engelbrecht, A. "Particle swarm optimization approaches to coevolve strategies for the iterated prisoner's dilemma". *IEEE Transactions on Evolutionary Computation*, 9, 562–579. 2005.

[31] Al-Haddad, R., & Sukthankar, G. "Psychologically-Inspired Agent for Iterative Prisoner's Dilemma". *Twenty-Third International Florida Artificial Intelligence Research Society Conference, 2010*, pp. 2-7 Florida: AAAI Press.

[32] Mittal, S., & Deb, K. "Optimal Strategies of the Iterated Prisoner's Dilemma Problem for Multiple Conflicting Objectives". *IEEE Transactions On Evolutionary Computation*, 13 (3), 554-565. 2009.

[33] Beaufils, B., Delahaye, J., & Mathieu, P. "Our Meeting With Gradual: A Good Strategy For The Iterated Prisoner's Dilemma". *Proceedings of the Artificial Life* V (pp. 202-209). Nara, Japan: Oxford University Press. 1996.