A Novel Meta-Heuristic Based on Soccer Concepts to Solve Routing Problems

Eneko Osaba
Deusto Institute of Technology
University of Deusto
Av. Universidades, 24
Bilbao, Spain
e.osaba@deusto.es

Fernando Diaz
Telecommunications Dept.
University of Deusto
Av. Universidades, 24
Bilbao, Spain
fernando.diaz@deusto.es

Enrique Onieva
Deusto Institute of Technology
University of Deusto
Av. Universidades, 24
Bilbao, Spain
enrique.onieva@deusto.es

ABSTRACT

In this paper, we describe a new meta-heuristic to solve routing problems. This meta-heuristic is called Golden Ball (GB), and it is based on soccer concepts. To prove its quality we apply it to the Vehicle Routing Problem with Backhauls (VRPB) and we compare its results with the results obtained by a basic Genetic Algorithm (GA) and an Evolutionary Algorithm (EA).

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem solving, Control Methods and Search—*Heuristic methods*

General Terms

Algorithms

Keywords

Meta-heuristics, Golden Ball, Genetic Algorithm, Routing Problems, Vehicle Routing Problem with Backhauls.

1. INTRODUCTION

Today, routing problems are widely studied in artificial intelligence, which are subject of a large number of works every year. There are a lot of problems of this type, the Traveling Salesman Problem [2] or the Capacitated Vehicle Routing Problem [4] are two of the most studied. The interest for the resolution of these problems lies in its complexity and in their applicability to real life. Being NP-Hard [3], their scientific interest makes that many studies focus on solving them, using a wide variety of techniques.

The objective of this paper is to present a new meta-heuristic based on soccer concepts for solving routing problems. This new technique is a multiple population-based algorithm, and we called Golden Ball (GB). It divides the different solutions of the problem in different teams, which improve independently and cooperatively and face each other in a competition. This competition will be crucial to decide the transfer of solutions between teams and to decide the model of training of each team.

To prove the quality of our new meta-heuristic and to demonstrate that it is a good alternative to solve routing

problems, we have used it to solve the well-known Vehicle Routing Problem with Backhauls (VRPB) (see [5] for further information about the problem) and we have compared the results obtained by our algorithm with the results obtained by the well-known Genetic Algorithm (GA) [1] and an Evolutionary Algorithm (EA) only based on mutations.

2. GOLDEN BALL META-HEURISTIC

As we said in the introduction of this work, our new meta-heuristic is a multiple population based algorithm which takes some concepts related to soccer for the search process.

The first step is to create a set of solutions, called P, which will make up the initial population. Every solution p_i is called *player*. All the players are created randomly, and once created, they are randomly distributed among the different teams t_i that will form the system. This division is done iteratively, obtaining a player p_i from P and inserting into t_i until reach PT, which marks the number of players per team. The number of teams is defined by the parameter TN.

The quality of each player p_i is represented by a real number q_i. This number is determined by a cost function $f(p)$, which depends on the problem. Each t_i has a p_{cap}, which is the player with better q_i of his t_i. To calculate the strength TQ_i of each t_i, the meta-heuristic takes into account the q_i of all the p_i that comprise it. TQ_i could be expressed by the following formula, taking into account that q_{ij} is the q of the player i of the team j:

$$TQ_i = \sum_{i=0}^{PT} q_{ij}/PT$$

2.1 Central phase

In this central step, the teams train independently or cooperatively, and they improve their power little by little. Meanwhile, the teams face each other creating a league competition that helps to decide the transfer of players from different teams. This process is divided into seasons (S_i). Each season has two periods of player transfers. A S_i also has as many matches as necessary to complete a conventional league, where all teams face each other twice. Finally, a S_i has as many training sessions as matches in the season.

The training phase is where all the players from each team receive a training session that makes them improve. Each team has a different training method, namely, a successor

function that works on a particular neighborhood structure in the solution space. This training method is assigned randomly at the initialization process. For each training, this function is applied a certain number of times to improve a p_i. The p_i' generated is accepted only if $q_i' > q_i$. This way, each team examines in different manner the neighborhoods of the players it possesses, making the players evolve in a completely different way, depending on the team on which they are.

Another kind of training is that we called *Custom Training*. It may happen that a player p_i is in a period in which, despite receiving training, it does not experience any improvement in its q_i. From the viewpoint of optimization, this happens when p_i is in a local optimum. This way, these trainings are performed by p_i with the help of the p_{cap} of his t_i, which is the player with the best quality in the team. With these trainings, p_i can escape from a local optimum. From a practical standpoint, a custom training is a combination of the characteristics of these two teammates, resulting in a new player who has probably taken a leap into the solution space. This jump can be beneficial to the search process, as it can help a more thorough exploration of the solution space.

After the training phase, matches are performed, and as in real world, they are between two teams. Matches are created as needed to complete a league, considering that all teams have to face each other twice in a season. Each match consists of PT chances. Each chance materializes in goal through a tournament between two p_i for each team, which are faced by his team position. The player with higher q_i wins the chance and it supposes a goal for his team. As in real life, the team which wins the match, obtains 3 points, and the loser gets 0 points. In case of a tie, each team gets 1 point. The points scored by each team are used to perform a classification, sorted by the number of points scored, being the best the team that has won more points. This classification is decisive in the period of transfers.

This period of transfers is a process in which the teams exchange players between them. In the middle and the end of each S_i, the t_i that are in the top half of the classification of the league are reinforced with the best p_i of the teams of the bottom half. While the teams in the lower half will have to settle with the acquisition of the less good p_i of top teams. These interchanges of p_i help the search process of the meta-heuristic. They allow the different treatment of the solutions during the execution, avoiding falling easily into local optima and increasing the searching capability of the technique.

Teams can also change their training method, and it is made as follows: In each period of transfers, all t_i from the bottom half of the table change their training function, hoping to get another training method which improves the TQ_i of the team.

Finally, the execution finishes when the sum of the powers of all the teams does not improve comparing to the previous season.In this moment, the algorithm returns the q_i of the best player of the system as the solution of the problem.

3. EXPERIMENTATION

In this section we will show the results obtained by applying the GB meta-heuristic to the VRPB and we will compare this outcomes with the results obtained with a basic GA and an EA. For both algorithms we have used the same functions and the same parameters (eg, the population size), so that the difference between them is the structure and the manner of working with the solutions and functions.

Table 1 shows the results obtained by the three algorithms. Tests were performed on an Intel Core i5 - 2410 laptop, with 2.30 GHz and a RAM of 4 GB. For each instance we display the total average and the standard deviation. The number of executions for each instance is 50. First 6 instances were obtained from the VRP Benchmark of Solomon, while the remaining 5 were obtained from the CVRP set of Christofides and Eilon. All instances are picked up at the VRP Web (*neo.lcc.uma.es/vrp*) and the types of demand have been changed to have pickups and deliveries.

Regarding GB parameters, TN used for these tests is 6, while PT is 12. The trainings number without improvement for a custom training is 5.

Table 1: Results of the algorithms for the VRPB

Inst.	GB	GA	EA
C101	631.24 ±32.2	688.93 ±46.6	702.14 ±38.5
C201	619.04 ±20.6	800.49 ±93.4	689.57 ±52.2
R101	868.43 ±19.6	902.83 ±24.1	932.87 ±38.7
R201	999.99 ±24.6	1084.85 ±67.8	1103.24 ±75.5
RC101	554.73 ±16.2	616.18 ±21.2	604.54 ±22.5
RC201	1105.73 ±29.9	1211.34 ±76.7	1225.51 ±65.5
Eil30	503.88 ±14.9	592.60 ±33.1	580.14 ±25.8
Eil33	768.57 ±28.9	817.78 ±30.1	815.48 ±31.2
Eil51	626.03 ±14.4	653.61 ±22.4	675.84 ±25.5
EilA76	824.20 ±23.7	863.16 ±23.7	856.26 ±21.6
EilA101	1097.90 ±38.5	1112.55 ±36.4	1186.46 ±45.8

4. GENERAL CONCLUSIONS

In this paper we have presented a new population based meta-heuristic for solving routing problems. After seeing the results, we can say that this new meta-heuristic is a good alternative to solve routing problems, being comparable with the GA and the EA, at least for the VRPB. As future work, we can mention the intention of applying this new meta-heuristic to a real environment, making a more elaborate objective function and creating more complex constraints.

Finally, the algorithm proposed in this work will be part of the PRODIS project (Grant PI2011-58, funded by the Basque Government in Spain).

5. REFERENCES

[1] D. E. Goldberg. *Genetic algorithms in search, optimization, and machine learning.* Addison-Wesley Professional, 1989.

[2] E. L. Lawler, J. K. Lenstra, A. R. Kan, and D. B. Shmoys. *The traveling salesman problem: a guided tour of combinatorial optimization*, volume 3. Wiley New York, 1985.

[3] J. K. Lenstra and A. Kan. Complexity of vehicle routing and scheduling problems. *Networks*, 11(2):221–227, 1981.

[4] P. Toth and D. Vigo. Models, relaxations and exact approaches for the capacitated vehicle routing problem. *Discrete Applied Mathematics*, 123(1):487–512, 2002.

[5] P. Toth and D. Vigo. Vrp with backhauls. *The Vehicle Routing Problem, SIAM Monographs on Discrete Mathematics and Applications*, 9:195–221, 2002.