An Application of a Discrete Firefly Algorithm in the Context of Smart Mobility

Ezzeddine Fatnassi(✉), Noor Guesmi, and Tesnime Touil

Institut Supérieur de Gestion de Tunis, Université de Tunis, 2000 Le Bardo, Tunisia
ezzeddine.fatnassi@gmail.com

Abstract. Firefly algorithm (FA) is a swarm intelligence based optimization method. It is based on the social behavior of fireflies where the brighter firefly attracts the less brighter one. In this paper, we present a discretization of FA in order to solve a problem related to the context of smart mobility. First, the context of smart cities and smart mobility is presented and a related optimization problem is proposed. Then, a discretization of FA to solve the problem is proposed. The proposed algorithm is based on transforming the FA functions into a discrete ones capable of manipulating a permutation of integers. Computational experiments on a set of instances from the literature demonstrate the efficiency of the proposed methodology.

Keywords: Smart cities · Smart mobility · Firefly algorithm · Electric vehicle

1 Introduction

Nowadays, the world is evolving at an incredible pace. In fact and in 2008, the United Nation stated that around 50% of all people are living in urban areas. Forecasts also show that in 2050 70% of the total population will live in urban areas. Consequently, cities centers are becoming the hub and the center of the physical (telecommunication and technologies) and human (qualified staff) capitals as well as commerce. This is due to the transformation observed by the world economy as it is becoming more integrated and service based. Cities are also considered as the majors centers for resource consumption. It should be noted that urban areas are responsible for 75% of the world's energy consumption and 80% of the greenhouse emission.

These features contribute on creating new urban environment characterized by the growing demand for the increase of the urban's quality of life as well as a sustainable and efficient development and an efficient use of the resources. Consequently, new ways of managing cities need to be considered in order to optimize resources [15].

Based on this context, research in smart cities and Intelligent Transportation Systems is needed in order to solve problems related to urban areas in a smarter ways.

R. Silhavy et al. (eds.), *Artificial Intelligence Trends in Intelligent Systems*,
Advances in Intelligent Systems and Computing 573, DOI 10.1007/978-3-319-57261-1_43

Smart city can be defined as the city that uses Information and Communications Technology (ICT) in order to offer to its citizen an interactive and efficient infrastructure and utilities. Consequently, ICT is considered as a key factor for the transition from basic infrastructure to a more connected and sustainable infrastructures enabling a smart city.

There are six most-common pillars of smart cities: (i) smart economy, (ii) smart people, (iii) smart governance, (iv) smart mobility, (v) smart environment, and (vi) smart living [7]. There are different ways of enabling a city to be smart. Among them one could consider the implementation of new services based on city's priorities. The implementation of such services has to take into account not only local regulation and priorities but also the citizens preferences as they are the main target of such services. In this paper, we focus on the reduction of the total travel time of citizens (Smart Mobility) and the reduction of greenhouse gas emission (Smart Environment). These two elements of the smart cities concept fuel the need for the implementation of public and smart urban transportation tools that reduce the impact of transit options on the environment.

Therefore, it comes the need for a novel and sophisticated transportation system that is efficient in term of ecological and societal aspect that could improve and overcome many issues related to urban mobility of goods and persons. Generally in smart cities and smart mobility, there are two ways to deal with sustainability issue:

– Enhance the quality and the performance of already existing urban movers' tools.
– Invest in a completely new and enhanced transportation system that is more ecological and energy efficient than the traditional and conventional transportation tool.

In this paper, we focus on the second option as we aim to introduce a relatively new transportation system within the concept of smart cities namely Personal Rapid Transit (PRT). PRT is an innovative transportation tool as it operates a set of electric driverless vehicles over a network of dedicated guideways.

In order to minimize its total energy consumption a discrete Firefly Algorithm is proposed to tackle a static deterministic routing problem related to PRT.

The proposed PRT problem is an optimization one which is proven to be NP-hard. Hence, it is difficult to be solved using a traditional exact method within small computational time which represents the requirement for such a solution approach in a highly dynamic context. Therefore, it comes the need for an effective algorithms to tackle the proposed problem. In the recent years, several new optimization algorithms inspired by swarm intelligence have been proposed e.g. ant colony optimization (ACO) [14], artificial bee colony (ABC) [8] and Firefly Algorithm (FA) [16].

FA is a bio-inspired optimization algorithm that mimic the social behavior of fireflies. Fireflies use light in order to attract other mates. Attraction among fireflies determines their movements. The attractiveness is related to the brightness of fireflies. Since its introduction, FA has been applied to several optimization problems such as nonlinear and non-convex optimization problems [16,17].

FA is implemented initially for solving continuous optimization problems. In case we want to apply the FA to a discrete problem such as the PRT, modifications need to be made on the original FA in order to make it able to solve the proposed PRT problem.

In this paper, we propose a FA variant that simulates the behavior of fireflies behavior by implementing discrete versions of the different components of FA. Consequently, the main contributions of this paper can be summarized as follows. First, a discrete version of FA is proposed in order to be applied in the context of smart mobility and green transportation. Second, the computational time of the proposed algorithm is analyzed. Third, the proposed approach is shown to effectively reduce the total travel time of PRT system.

In this paper, a methodology for modeling a static problem related to PRT will be presented. Then, a discrete FA is proposed to tackle this problem. Next, the computational results are described. Finally, we provide some conclusions and perspectives.

2 PRT Problem Definition and Modeling

Integrating PRT into a smart city would be very beneficial in the context of smart mobility and smart environment. In fact, the system has the possibility to offer several advantages over classical transportation tools in order to reduce total travel time, congestion and carbon emissions. In the literature, several studies [5] already show that the total trip time and carbon emissions using PRT system are less than several public and classic transportation tools.

Based on the specificities of PRT as a potential improvement in the context of smart cities, we could note that it is important to focus on the economic and environmental urban context while implementing such an innovative system. In fact, the PRT offers an on-demand transportation service which results necessarily in a set of empty vehicles displacements. More specifically as the demand ending at a specific station does not necessarily equal to the demand starting from that same station, the PRT system could generate a set of displacements where vehicles move empty in order to take specific passengers from a station. This set of empty moves engenders an economic loss as well as contributing on harming the environment as the energy used for the empty displacement was simply wasted and not used efficiently. So it could be interesting from an operational point of view to enhance its efficiency by reducing its empty moves while making it more sustainable and economically reliable.

Consequently, we propose in the next to solve a static routing problem related to PRT [12]. Let us suppose to have a fully connected PRT network with a set of M stations and one depot named D. We focus in this paper exclusively on the operational use of PRT.

Consequently, we suppose to have a list of PRT' transportation demand T that need to be satisfied. Each demand $i \in T$ is characterized by a quadruplet: (i) DT_i: depart time, (ii) AS_i: arrival station, (iii) DS_i: depart station, (iv) AT_i: arrival time which is the is the depart time DT_i plus the duration of the trip with shortest path, between the depart and the arrival stations.

We suppose also to have a set of unlimited number of vehicles initially located in D. Each vehicle has a limited battery capacity denoted B that make it run for a limited determined time. PRT batteries could only be charged at the depot. We suppose also to have a matrix named Sp representing the shortest path between each pair of stations.

The proposed problem is modeled based on a graph $G = V, E$. The set of nodes is represented by V. Each node in V represents one PRT' transportation demand. Two dummy nodes s and t are added to V which represent the depot. $V^* = V/\{s,t\}$ Let us define also E as the set of arcs. $E^* = E/\{(i,j); i = s$ or $j = t\}$ where s and t are two dummy nodes representing the depot. For each pair of nodes i and j, we define an arc based on the following rules:

– for each i and $j \in V^*$ and $AT_i + Cost_{(AS_i, DS_j)} \leq DT_j$ an arc (i,j) is added with a cost c_{ij}, equals to the travel time needed to move from AS_i to DS_j in addition to the travel time needed to move from DS_j to AS_j
– We add a set of arcs connecting the depot node s to each node in V^*. The cost of this arc is equal to the travel time needed to reach the arrival station of node i from the depot while passing by the departure station of node i.
– for each node $i \in V^*$, we add an arc (i,t). The cost of this arc is the travel time needed to reach the depot starting from the arrival station of trip i.

The objective of our problem is to find the set of roads starting and ending at the depot while covering all the trip nodes exactly once and respecting the battery capacity of the PRT' pods. Based on this problem modeling, we could note that our problem is assimilated to the asymmetric distance constrained vehicle routing problem (ADCVRP). The ADCVRP is the problem of finding a set of vehicles'roads starting and ending at the depot with respect to a maximum allowable distance constraint. The ADCVRP is proven to be NP-Hard [1]. This theocratical problem was studied in the literature in the work of Laporte et al. [9] and in the work of Almoustafa et al. [1]. The high complexity of this problem (NP-Hard) makes it suitable to be solved using a bio-inspired algorithm [6]. In the next, we focus on adapting and applying a discrete fire flies algorithm for solving the proposed problem.

3 Firefly Algorithm

3.1 Basic Firefly Algorithm

Firefly algorithm was developed first by Xin-She Yang [16]. FA is like the particle swarm optimization (PSO) algorithm. It is a stochastic based search algorithm which manipulates a population of firefly. Solution of the proposed problem is represented by a single firefly in the population. Fireflies generally move to other positions in the search space in order to find potential new solutions. Attractiveness between different fireflies is determined by the intensity of the emitted light. The emitted light is proportional to the fitness value of the firefly.

Algorithm 1. Basic Firefly Algorithm

Initialize the population of Fireflies of size n
while (termination criterion is not met) **do**
  for ($i = 0$ to n) **do**
    for ($j = 0$ to n) **do**
      if $F(x_i) > F(x_j)$ **then**
        Move x_i toward x_j
        $d_{ij} \leftarrow \text{Distance}(x_i,x_j)$
        $\beta \leftarrow attractivness(I_0, \gamma, d_{ij})$
        $x_i \leftarrow (1-\beta)x_i + \beta x_j + \alpha(Random() - \frac{1}{2})$
        Compute the fitness value of the new x_i.
      end if
    end for
  end for
  Best firefly moves randomly
end while

The attractiveness of a firefly is attenuated over the distance. The basic form of FA is presented in Algorithm 1.

In Algorithm 1, n is the size of the population. I_0 is the intensity of light at the source. α represents the size of the random step and γ represents the absorption coefficient.

3.2 FA Discretization for PRT Problem

Based on Algorithm 1, it is clear that several functions related to FA need to be re-implemented in order to be able to solve the PRT problem such as the steps of movements of the fireflies as well as the attraction function.

To do so, solutions (the set of fireflies) in our algorithm are represented by a permutation of trips and the algorithm manipulates them accordingly. To evaluate the different solutions in our algorithm, a cost function from the vehicle routing problem (VRP) literature is adapted. In fact, we use the Split function based on the work of Prins [13]. This cost function and based on a permutation of customers in VRP could find a set of minimum cost routes. An auxiliary graph is constructed where each node represents a VRP customer and each edge represents a feasible route based on the permutation of customers. Then, the Bellman algorithm [2] is used in order to find the least cost set of routes. The same principle is applied in our algorithm where a solution is represented by a set of trips instead of a set of customers. More details could be found in [13].

3.3 Initial Fireflies

In the FA version proposed by Xin-She Yang [16], the initial fireflies are generated randomly in order to cover the whole search space in an uniform distribution. Several approach in the literature exists in order to generate initial solutions for optimization problems such as greedy approaches. However and in this paper, we propose to use random solution'generation as using this method would guarantee to have a scattered initial population over the search space and therefore a more diverse fireflies. We added to the initial population one enhanced firefly based on the constructive heuristic proposed by Mrad et al. [11].

3.4 Distance Function

In the literature, there is different ways to compute the distance between two distinctive solutions:

- The Hamming distance which corresponds to the total number of trips minus the number of trips that exist in the same position on the two solutions
- The Swap distance which corresponds to the number of swaps needed to move from the first to the second solution.

However, it is clear that the Hamming distance is proportional to the difference in the objective function between the two compared solutions. Consequently and based on this observation, we compute in this paper the difference of the objective function using the Split function [13] in order to measure the distance between two different solution.

3.5 Attraction Function

The implementation of the attraction between two different fireflies represents the one of the most important feature to be implemented in order to apply the FA in the context of smart mobility. The FA was initially implemented for continuous optimization. Therefore, the attraction function needs to be implemented in a way to respect its original implementation for continuous optimization.

The original FA algorithm and in order to compute the next position of a firefly use the following function:

$$x_i \leftarrow (1-\beta)x_i + \beta x_j + \alpha(Random() - \frac{1}{2}) \tag{1}$$

Equation 1 could be divided into two subfunctions

$$x_i \leftarrow (1-\beta)x_i + \beta x_j \tag{2}$$

and

$$x_i \leftarrow x_i + \alpha(Random() - \frac{1}{2}) \tag{3}$$

Therefore and in order to implement a discrete version of FA, we need to implement a search versions of the Eq. 2 (which we call the β step) and 3 (which we call the α step).

3.6 The β Step

The β step should be implemented in such a way to bring one firefly closer to another firefly. Consequently and based on this principle, the distance between two solutions should be reduced after applying the β step. Let us recall that we are using the difference between the fitness of two solutions as a measure of their relative distance. Therefore and in order to implement the β step, the set'size of common trips between the two solution needs to increase. Consequently and as a first step, we need to extract the set of the common trips between the two solutions and insert it into the output solution. Figure 1 shows an example of this first step.

Next, we need to fill the gaps in the output solution using the rest of missing trips from the two input solutions i and j. Therefore, we need to iterate on the gaps in the resulting solution and choose whatever to insert a trip from the solution i or j. We need also to ensure that there is no redundant trips in the resulting solution as we are working on a permutation as a representation of the solutions in the FA. This could be done using a probability $\beta = \frac{1}{1+\gamma d_{ij}}$ where d_{ij} is the distance between solutions i and j and γ is a parameter for the FA which defines the amount of randomness in the selection between the two solutions. Finally, we need to fill the gaps in the resulting solution in a random order to guarantee a minimum of diversity in the output solution. An illustrative example is shown in Fig. 1. In the first step, we copied the trip from solution j in order to fill the gap at position 6. Next, we copied a trip from solution i in order to fill the gap at position 2. In the third step, we did not copy any trips at position 5 as the two trips from solutions i and j are already used in the output solution.

Finally in the β step, we need to fill the last gaps in the output solution with the missing trips in a random order in order to obtain a valid solution. Therefore and based on the example shown in Fig. 1, the only missing trip is trip 8. Consequently, we insert it in the remaining gap in the output solution.

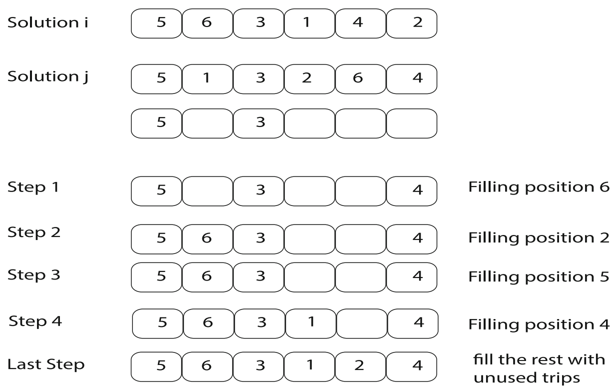

Fig. 1. Illustrative example of the β Step.

3.7 The α Step

The α step is implemented in this paper as a shift to move to one of the neighboring solution of its input solution. Therefore, the α step could be implemented in a variety of ways. However, we should keep in mind that the result solution of the α step needs also to mimic the attraction process of the fireflies. In this paper and to do so, we applied several random swaps in the input solution in order to obtain the new generated firefly. Using swaps as neighborhood operator guarantees to generate a solution with the minimum distance between the input and output solution.

4 Experimental Results

We now present the computational study performed to investigate the results of our FA for solving the proposed optimization problem. An analysis of the obtained results is also performed to prove the efficiency of our algorithm.

4.1 Test Instances

The FA is implemented using a personal computer with an Intel i5 3.2 GHz CPU and 8 GB of RAM, and running the Microsoft Windows 7 operating system. Since we are treating a PRT routing problem, we adapted instances from the literature to our context in order to test the FA [3]. The instances used in this paper are based on the number of trips to serve which varies between 10 and 100 trips in a step of 5. For each size of trips, we used 40 instances. For more details about the PRT instances and the instance generator the reader is refereed to [12].

4.2 Computational Results

The computational results of the FA are exposed in Table 1. The obtained results are compared against a valid lower bound values (LB) [4] taken from the literature for the proposed problem using the GAP metric which is computed as follows.

$$\text{GAP} = (\frac{(SOL - LB)}{LB}) \times 100$$

Table 1 shows that the proposed FA obtains good quality results for the proposed optimization problem. More specifically, the FA founds an average gap of 2.621%. Also, we could note that the average GAP varies between 0.608% and 5.228%.

These results shows that applying FA in the context of smart mobility is capable of producing satisfying results for our problem. We also point that the obtained results are compounded by the fact that the FA founds its results in a small computational time (3.555 s). In fact and by performing a rather simple and straightforward rules, the FA is capable of performing an intensive global search over the search space. Also, we should note that the performance of our proposed method against the proposed method of Mrad et al. [11] shows an enhance of performance by 3.28%.

Table 1. Results of the FA

Instance size	Average gap %	Average time sec
10	1.105	1.421
15	0.799	1.509
20	0.608	1.852
25	0.964	2.005
30	0.998	2.221
35	1.702	2.496
40	1.759	2.622
45	1.718	2.662
50	1.964	2.878
55	3.123	3.446
60	2.606	3.705
65	2.932	3.918
70	4.119	4.105
75	3.530	4.618
80	4.154	4.754
85	4.332	5.119
90	3.713	5.636
95	4.442	6.554
100	5.228	6.020
Average	2.621	3.555

5 Conclusions and Perspectives

In order to enhance the smart cities performance, we focused in this paper on enhancing the performance of urban transportation tool (smart mobility) while reducing the carbon emission (smart environment). We proposed in this work to use PRT vehicles as a smart mobility tool in the context of smart cities. Next, we focused on implementing a FA for solving an optimization problem related to PRT. The problem consisted on satisfying a set of on-demand origin-destination pairs subject to several constraints such as maximum allowable distance constraint and time window constraint. As the FA is implemented for solving continuous optimization problems, we focused in this paper on the discretization of FA in order to make it able to solve the proposed smart mobility problem. We reported in this paper extensive computational tests on a set of carefully generated instances taken from the literature. The proposed algorithm was shown to get good quality results. As a future work, we are working on the concept of Physical Internet [10] in order to enable hyper-connected transportation service in a sustainable way.

Acknowledgement. The authors would like to thank Dr. Hadhemi Achour for her valuable comments and suggestions to improve the quality of the paper.

References

1. Almoustafa, S., Hanafi, S., Mladenović, N.: New exact method for large asymmetric distance-constrained vehicle routing problem. Eur. J. Oper. Res. **226**(3), 386–394 (2013)
2. Bellman, R.: Dynamic programming and lagrange multipliers. Proc. Natl. Acad. Sci. **42**(10), 767–769 (1956)
3. Chebbi, O., Chaouachi, J.: Reducing the wasted transportation capacity of personal rapid transit systems: an integrated model and multi-objective optimization approach. Transp. Res. Part E: Logistics Transp. Rev. (2015)
4. Fatnassi, E., Chaouachi, J.: A comparison of different lower bounding procedures for the routing of automated guided vehicles in an urban context. Int. J. Appl. Nonlinear Sci. **2**(1–2), 120–135 (2015)
5. Fatnassi, E., Chaouachi, J., Klibi, W.: Planning and operating a shared goods and passengers on-demand rapid transit system for sustainable city-logistics. Transp. Res. Part B: Methodol. **81**, 440–460 (2015)
6. Fatnassi, E., Chebbi, O., Chaouachi, J.: Discrete honeybee mating optimization algorithm for the routing of battery-operated automated guidance electric vehicles in personal rapid transit systems. Swarm Evol. Comput. **26**, 35–49 (2016)
7. Janecki, R., Karoń, G.: Concept of smart cities and economic model of electric buses implementation. In: Mikulski, J. (ed.) TST 2014. CCIS, vol. 471, pp. 100–109. Springer, Heidelberg (2014). doi:10.1007/978-3-662-45317-9_11
8. Karaboga, D., Kaya, E.: An adaptive and hybrid artificial bee colony algorithm (aabc) for anfis training. Appl. Soft Comput. **49**, 423–436 (2016)
9. Laporte, G., Desrochers, M., Nobert, Y.: Two exact algorithms for the distance-constrained vehicle routing problem. Networks **14**(1), 161–172 (1984)
10. Montreuil, B.: Toward a physical internet: meeting the global logistics sustainability grand challenge. Logistics Res. **3**(2–3), 71–87 (2011)
11. Mrad, M., Chebbi, O., Labidi, M., Louly, M.A.: Synchronous routing for personal rapid transit pods. J. Appl. Math. **2014**(1), 1–8 (2014)
12. Mrad, M., Hidri, L.: Optimal consumed electric energy while sequencing vehicle trips in a personal rapid transit transportation system. Comput. Ind. Eng. **79**, 1–9 (2015)
13. Prins, C., Lacomme, P., Prodhon, C.: Order-first split-second methods for vehicle routing problems: a review. Transp. Res. Part C: Emerg. Technol. **40**, 179–200 (2014)
14. Samà, M., Pellegrini, P., D'Ariano, A., Rodriguez, J., Pacciarelli, D.: Ant colony optimization for the real-time train routing selection problem. Transp. Res. Part B: Methodol. **85**, 89–108 (2016)
15. Trilles, S., Calia, A., Belmonte, Ó., Torres-Sospedra, J., Montoliu, R., Huerta, J.: Deployment of an open sensorized platform in a smart city context. Future Gener. Comput. Syst. (2016)
16. Yang, X.S.: Firefly algorithm, stochastic test functions and design optimisation. Int. J. Bio-Inspired Comput. **2**(2), 78–84 (2010)
17. Yang, X.S., Hosseini, S.S.S., Gandomi, A.H.: Firefly algorithm for solving non-convex economic dispatch problems with valve loading effect. Appl. Soft Comput. **12**(3), 1180–1186 (2012)