COLLECTIVE COMPUTATION: TURNING THE UNDERGROUND INTO AN ANT NEST

Clemencio Morales, Luis Fernando de Mingo

***Abstract**: The management and proper use of the Urban Public Transport Systems (UPTS) constitute a field as critical as little investigated according to its relevance and urgent idiosyncrasy within smart cities realm. In this paper, a newfangled approach by using the Natural Computing paradigm and Collective Computation is shown, more concretely taking advantage of an Ant Colony Optimization algorithm variation in order to build a system that makes the complete control of the UPTS a tangible reality.*

***Keywords**: Smart City, Natural Computing, Collective Computation, Urban Public Transport System, Wireless Sensor Networks*

ACM Classification Keywords: *10003120.10003138: Human-centered computing - Ubiquitous and mobile computting, 10010147.10010178: Computing methodologies - Artificial intelligence, 10010147.10010257.10010293.10011809: Computing methodologies - Bio-inspired approaches, 10010405.10010481.10010485: Applied computing - Transportation*

***MSC**: 68Q32 Computational learning theory, 68T05 Learning and adaptive systems*

Introduction

Since its pioneer conception at 1843, the underground trains have changed in such a mastodontic number of ways that it will take ages to enumerate them. However, these changes have not been applied to the management system and conception of the underground itself as it is nowadays. On the one hand, the rapidly-growing massification of the world´s urban cores, in collaboration with the underground intensive use by citizens, is pushing the transition of these cores to the smart city purest concept, where every single element within the city has ratiocination enough for it to be called *intelligent*. The aforementioned massification can be seen in Fig. 1.

On the other hand, it is important to note that this need has been outlined by organisms such as C.E.O.E (Spanish Confederation of Business Organizations). In fact, as described in [CEOE, 2015]:

"This frame of sustainability and efficiency that must involve the Smart Cities, has a direct relationship with other key areas, such as [...] the **efficient management of the mobility of people** [...] [Cities are lacking] Indicators for the collection appropriate measures [...] [Cities systems need] real-time knowledge about incidents, and an improved efficiency and management of the public transport"

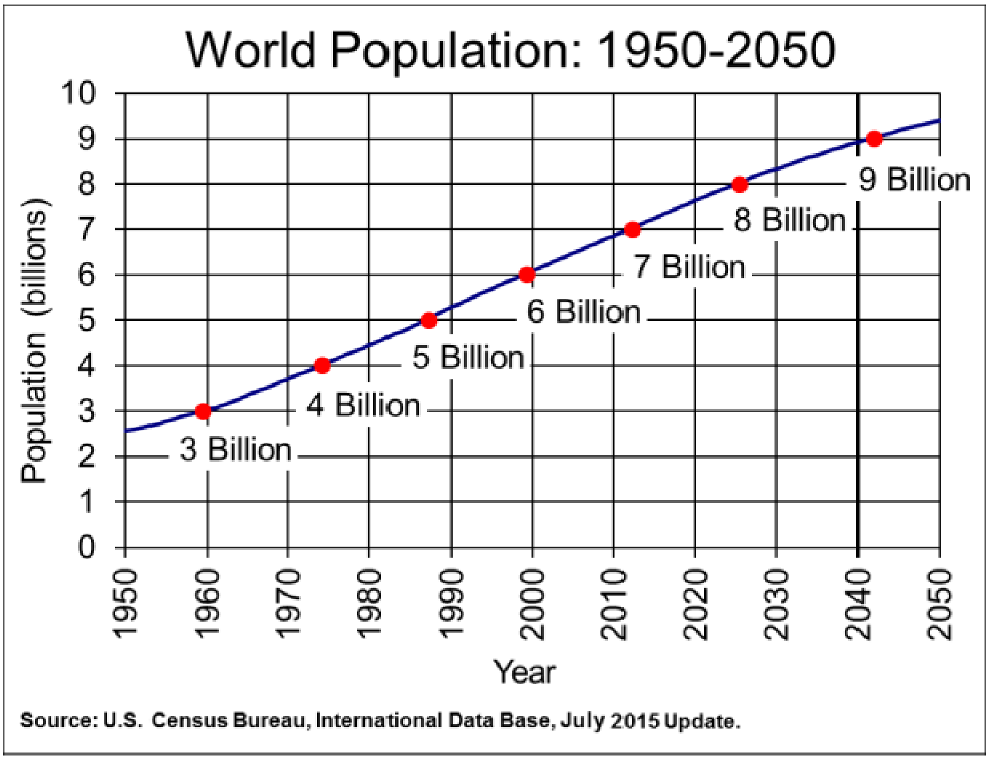

Figure 1. World Population estimated growth between the years 1950 and 2050
(Source: United States Census Bureau, International Data Base)

Even the concept of Smart City is still being under constant redefinition, most authors agree that many different individuals, agents and devices, operate with their environment within the Smart City realm. Therefore, as [Hollands, 2008] points out, the relation among all of these elements will define the behaviour of the Smart City itself. It is easy to realize that an important area of the Smart City will be based in the interaction between the different components of it with their environment. This fact disembogues in a **Socio-Collective Interaction**, where the smart city in general terms, and specially the underground system beneath, can be seen as a huge swarm, where agents collaborate between them. The aforementioned approach justifies the present project, mainly based on a change in the way of tackling the management processes of any underground system, using Collective Computation algorithms instead of the classical, graph-oriented ones.

Definition of the problem

The underground system beneath any urban core is a living, constantly-in-change entity. According the Annual Subway Ridership of the Metropolitan Transportation Authority [MTA, 2012], 3.410 billion itineraries were made last year within Beijing underground system. Thus, how can be these itineraries traced, letting the management know who is using the underground and when? How can the users rapidly know if there is an emergency or a path which is not working due to technical errors within the underground? How can we monetize the massive data that can be potentially generated by so many itineraries? The response to these questions is precisely the reason that justifies the ongoing project, which aims to create a synergy of elements achieved due to the application of many newfangled Computation Paradigms. These paradigms, in collaboration with a strict software of control purposes, that will operate with user´s Smartphone's, will head to an increase in the intelligence of the Urban Public Transport Systems (UPTS hereinafter).

Investigation goals

The present investigation has the objective of fixing, chiefly, the following goals, that define an accurate overview of the project investigation:

- Investigate the Computing Paradigms according to the realm of the Collective Collaboration: As it will be explained in further sections within this document, the Natural Computation stands as the best ally when it comes to this investigation aspect. This paradigm is made up by Genetic Algorithms, Ant Colony Optimization, Swarm Computing, Grammatical Evolution and Grammatical Swarm, which will be investigated in order to find possible improvements, if any.

- Find a nexus between the Computing Paradigms involved and the problem to solve: Once a strong theoretical overview has been given to the reader, the union point and nexus with the chased system will be described. It is remarkable that, at the time being, there is no application of these algorithms in the UPTS context, factor that increases the newfangled character of the present investigation.

- Design and development of a system that, using the needed paradigms within Natural Computation, allows a wide study of the behaviour of the underground users: In a nutshell, the system aims to become a tool that makes possible the study of the user´s behaviour, by taking dissociated data up in order to guard the privacy of the citizens. This objective will be possible thanks to the UPTS users Smart phones, for which an application is to be developed in the Android Operating System. Please note that this system will make possible to:

  * Make precise studies about the statistical population that uses the UPTS.

  * Know, in an accurate way, the most popular routes for the users, as well as their behaviour between the UPTS. It is remarkable that this factor constitutes an open door for an efficient[1] management within the system.

[1] It is important to note that the term efficient differs to effective in a subtle, but crucial manner; while an effective system achieves every objective, an efficient one achieves every objective as well but in the best, optimal way.

* Prepare, thanks to the estimations gathered from the statistical study of the data, the UPTS in order to deal with peaks. Such scenario can be predicted by attending at atypical values within the data set gathered by the system.

* Detect anomalous situations, such a blocked train within a tunnel, or different casuistry where the number of users standing at the platform is high enough to fear an accident, surpassing the capacity of the specific dock.

*Prepare alternative routes in case of intensive use and/or fault of the UPTS systems.

More functionalities and features, currently in an evaluation and delimitation stage.

Topology of the system

The system to be developed formed by essentially five groups of elements, has the main structure that can be seen in Figure 2:

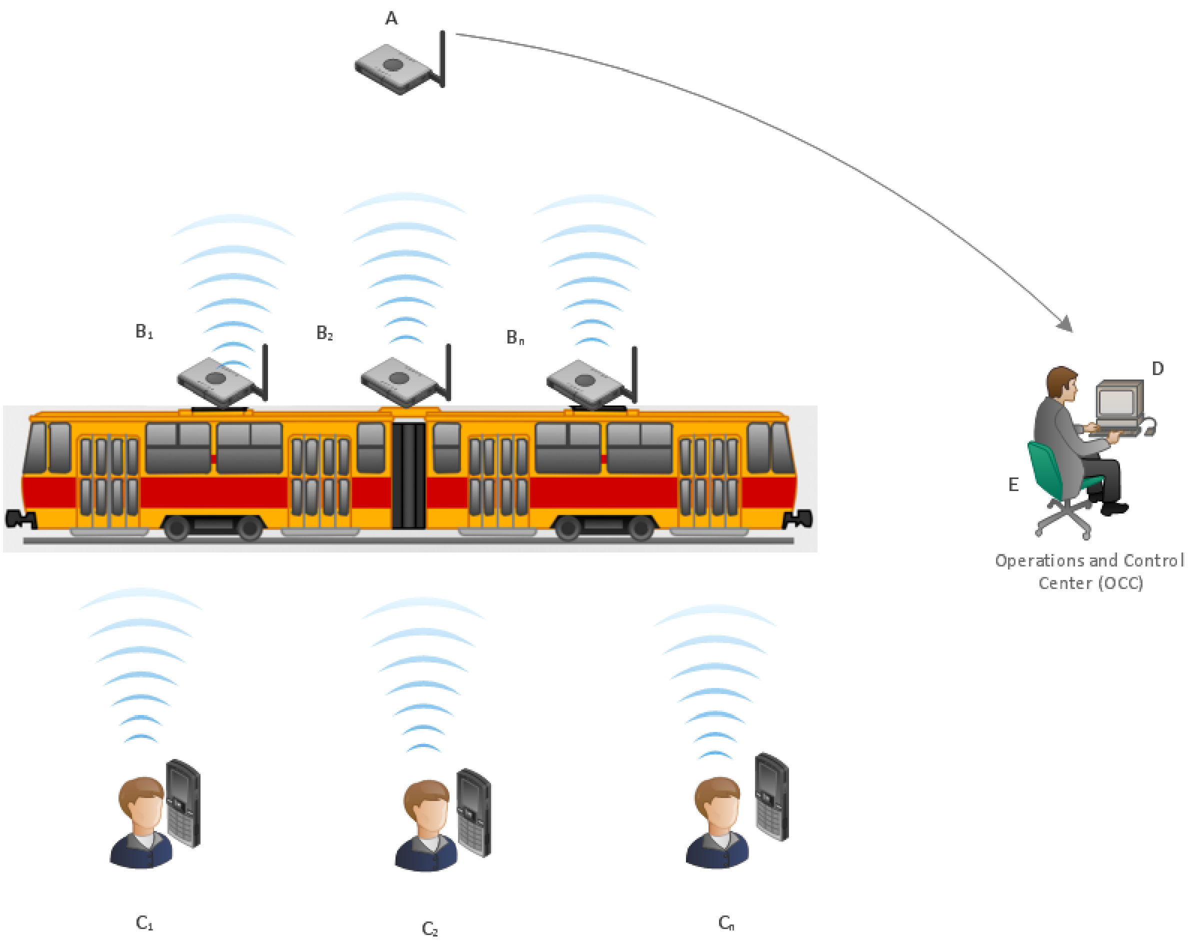

Figure 2. Overview of the currently-under-development system

As Fig. 2 shows, there are five groups of elements within the system, which are described below:

A: *Fixed Smart dust*: In Wireless Sensor Networks terms, this smart dust element, that will be unique in each UPTS station, will behave as the *sink* node. This special device will be integrated in the dock itself, receiving information from its B_n counterparts. The *sink* node will be the only element able to establish communication with the Operations and Control Center, *D*, and the elements within the *C* set.

B[1,2,3,…,n]: *Mobile Smart dust*: The present element of the system, embedded in the UPTS trains fleet, will behave as a *slave* of the *sink* smart dust. They will establish communication with the *sink*, and will receive data from the elements within the *C* set, which will be explained below.

C[1,2,3,…,n]: *Users smart phones*: This fundamental element within the system will be used tacitly by the users in order to make evident their presence in the dock. The elements within this group will be able to communicate with the *sink* smart dust, *A*, as well as with its counterparts in *B*, the *mobile smart dust* devices, that will make possible to know the number of users in the train. It is remarkable that the set formed specially by *A* and *B[1,2,3,…,n]* will shape the Wireless Sensor Network of the system, that will operate closely with the *C[1,2,3,…,n]* devices. A mobile application will be developed for the elements in *C*, that will retrieve the dissociated data of the users, let them know different routes in case of massive congestion, configure itineraries and show warning regarding abnormal situations that may occur within the UPTS.

D: *Operations and Control Center* (OCC): This element will behave as the management point within the system realm, receiving the data sent by the smart dust in every single station, showing the pertinent status and the presence, if appropriate, of abnormal situations from the safety/systems failure point of view.

E: *System Administrator*: Evaluator of the data showed by the OCC. Will operate accordingly to the UPTS current status and its environment, whether triggering a specific security protocol against failure or solving the different spurious scenarios that may occur.

Algorithms involved

Despite its apparent disparity, the following Computation Paradigms and the Algorithmic Techniques described below fall within the spectrum of the Natural Computation Paradigm. As long as the investigation is currently on an early-medium stage, the nexus with the system of some of these paradigms, as well as their application to the system, are still being under investigation.

As the accustomed reader will surely intuit, the algorithmic entities attached to this paradigm have, as its main base, the logic associated to phenomena present in nature, as well as the logic associated with the genetic-molecular base of the living beings, thus. As it can be read in [Rozenberg, 2012], we can formally define Natural Computing as the set of computing techniques that circumscribe to, at least, one characteristic defined within the following group:

* Obtain its base from observing nature, establishing a computing simile.

* Base its reasoning in the use of the computers in order to synthesize natural phenomena.

* Use natural materials, from the logic or physical point of view, like DNA strings or chromosomes, in order to achieve its computational processes.

Genetic Algorithms

As stated by Charles Darwin in his opus magnum *On the Origin of the Species* [Darwin, 1859], from immemorial times living beings have been forced to a continuous evolutive process looking for survival. Every single specie evolutes from a common antecessor looking for the adaptation to its environment and survive, following the process named *natural selection*. In a parallel way, *Genetic Algorithms* (GA hereinafter) follows the same pattern, trying to evolve a *population*. Thus, as it can be extracted from John H. Holland´s *Adaptation in Natural and Artificial Systems* [Holland, 1975], a GA can be formally defined as a set of ordered instructions, that aims to achieve an specific problem, which are based on the genetic-molecular base of the evolutive process of the living beings. It is remarkable that, despite the paternity of the GA is attributed to Prof. Holland, his sublime work means the colophon to the investigator cycle started by the distinguished Gregor Mendel in 1865, with his laws stated in *Experiments in Plant Hybridization* [Mendel, 1865], based on the investigation over *Pisum Sativum*. In his publication, Mendel describes, using this specific pea variation, the basic rules related to the characteristics transmission between individuals through genetic inheritance. Actually, a GA has the objective of evolution certain specimens that set a *population*. In order to chase this goal, the GA uses random operations that establish a simile with the natural processes related to biological evolution. These methods, called *genetic operators*, are the following:

- Selection: In this operator, the GA chooses individual genomes from the population in order to start a later breeding process. Selection can be made by means of various techniques, as seen in *A Comparison of Selection Schemes Used in Evolutionary Algorithms* [Blickle, 1996]. These techniques can be Roulette-Wheel Selection, Selection by Truncation, Selection by Ranking or Selection by Tournament, to quote a few of them.

- Crossover: Process whereby a variation in the chromosomes is done from a generation towards the following one. It is remarkable that, following the natural simile, the crossover mocks the sexual reproduction of the living beings. Letting a binary string be the information to be represented, there are several crossover techniques, and they all produce permutations in the chromosome. Seeing the chromosome as a set of alleles, the technique of *crossover in a point* can be an illustrative example; as shown in the following figure, once a bit within the chromosome is selected, every successive allele is exchanged between a chromosome and its pair, generating a new offspring in the process (Figure 3):

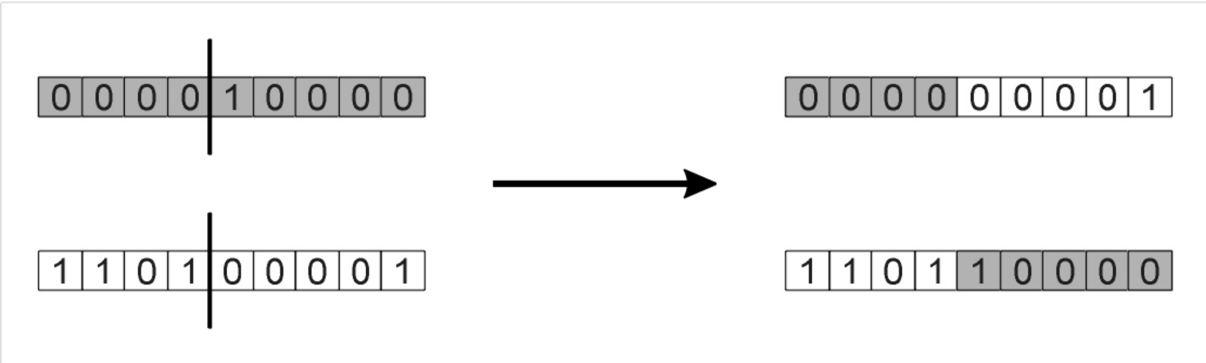

Figure 3. Genetic operator crossover in a point between one-byte alleles (Source: University of Amsterdam – Faculty of Sciences, Department of Computer Sciences)

- Mutation: Variation within the genotype of a living being. Represents the action of the mutagens present in the ecosystem. It is remarkable that the genetic unit able to mutate is the gene, atomic, inheritable unity of data that builds up an individual's DNA.

- Recombination: Process whereby a DNA portion is cleaved in order to provide its further union to a different genetic material molecule. It is important to note that this action provokes different genetic permutations in a specie regarding its predecessors, producing chimeric alleles. This advantage makes the sexual reproduction possible between living beings, while avoiding Muller´s ratchet1.

Ant Colony Optimization

As Marco Dorigo and Gianni Di Caro establishes several times along [Dorigo, 1992], Ant Colony Optimization (ACO hereinafter) is the name that refers to a multi-agent paradigm where every agent´s behaviour is inspired on the ants idiosyncrasy when searching for livelihood. The algorithms that fall within this classification are based in Goss Experiment, using an *Iridomyrmex humilis* colony. In this experiment, the ant nest is connected to a livelihood source by means of two different paths, where one is longer than its counterpart, as the following Figure 4 shows:

---

1 Named after its discoverer, Hermann Joseph Muller, is the process by which the different genomes of an asexual population accumulate deleterious mutations in an irreversible manner, that may result in the irrevocable extinction of the specie.

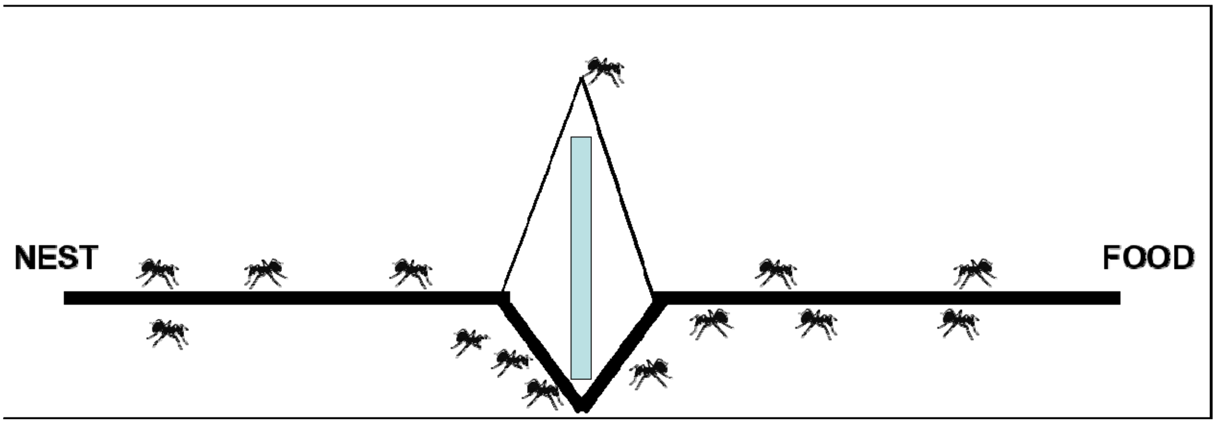

Figure 4. Goss Experiment representation (Source: BioDat Research Group - University of Karlovo, Prague)

After allowing the ants to freely move themselves along the scenario, it can be seen that, after an initial moments, they always choose the optimal, shorter path to the livelihood source. It is remarkable that, as well, this experiment demonstrates that a route selection probability is directly proportional to the length difference between both paths.

After studying the results thrown by Goss Experiment, a question arises; How do all the ants know what is the shortest path? The answer to this question is based on the concept known as stigmergy. The aforementioned concept alludes to those collaboration protocols, through the physical medium, where the different components collaborate due to the accumulation of objects or magnitudes in the environment, such pheromones or humidity. This concept is, precisely, the main tool within ant's communication; as the ants go back and forwards to the livelihood source, they deposit a chemical substance called pheromone. As it happens in several species, this substance provokes specific reactions and behaviour in the individual counterparts, allowing this, on this case, to know what the shortest path is.

It is remarkable that the directive that makes each ant *k*, placed in the *i-th* node, using a pheromone trail τ_{ij} in order to calculate the probability it has to use to chose a node *j* that belongs to *N*, as well as the following node where it has to move along, where N_i constitutes the set of nodes adjacent to *i*, is given by the formula:

$$p_{ij}^{k}\begin{cases}\tau_{ij}\ if\ j \in N_i \\ 0\ if\ j \notin N_i\end{cases} \quad (1)$$

Particle Swarm Optimization

Since the dawn of science, many scientific have been intrigued by a movement, as elegant as optimal, present in nature: The harmonious synchrony in bird flocks and fish shoals, where the individuals are able to move without even rub with each other, despite the hundreds, thousands of elements in certain cases, of individuals present in these sets. Thanks to scientific investigation, it has been demonstrated that, apart from this optimal movement, these animals present certain *swarm patterns* in their behaviour.

Concretely, it is important to highlight the hyperbolic interest of Grenander & Heppner on their opus magnum *A stochastic nonlinear model for coordinated bird flocks* [Grenander, 1990], where both zoologists synthesize their investigation referred to the nature-hidden directives that mark the asynchronous movement of the bird flocks, changing its direction suddenly in the presence of predators and tacitly regrouping, among other interesting abilities. In the same line, Reynolds *Flocks, herds and schools: a distributed behavioural model* [Reynolds, 1987] stands out, aiming to the study of the interesting choreography that birds deploy.

Clustering the aforementioned references as base, the Particle Swarm Optimization (PSO, hereinafter) paradigm is known as the technique that pretends to optimize a problem due to a meta-heuristic strategy, which is, due to the iterative *trial of improving* a candidate solution with regards to a pre-stipulated quality criteria. Thus, in a way that reminds to GA, PSO optimizes a problem starting from a set of candidate solutions, typically particles over the space, moving them along through the searching space without forgetting the premises of PSO mathematical base, which involves the position and the speed of the particles. As it can be inferred, the technique mimetizes the group behaviour of the aforementioned living beings, where each individual movement is influenced by the best *local* position known, while, in a parallel way, the *swarm* maintains a best *global* position known. This best global position is updated by the best position known by all the individuals in the swarm, fact that will guide the set to move searching for the best global position.

PSO adopts a tiny number of postulations along its execution process, exploring a mastodontic search space. Despite from that, PSO is a meta-heuristic, so it is not possible to adamantly ensure that the algorithm is going to find an optimal solution of the problem for every single case. In a more mathematical, accurate way, PSO does not use the *gradient* of the tackled problem, which means that this technique does not require the problem to be differentiable, as well as it happens in typical optimization methodologies such Cuasi-Newtonian methods or Gradient Descent. Thus, PSO can be used, enjoying a high success rate, in optimization problems that are especially non-regular, where there is certain ambient *noise*, or those presenting a dynamic, changing-over-time behaviour. PSO algorithm pseudocode can be stated as following, being S the swarm, *b_low* and *b_up* the preselected ranges and ω, φ parameters to be set:

```
for each (particle within S){
   position = generateRandomValue(S[i], b_low, b_up);
   position = bestKnownPositionByParticle(S[i]);
   if( f(p) < f(g) ){
      bestGlobalPosition = position;
   }
   speed = generateRandomSpeed(b_low – b_up, b_low - b_up)
 }

while(!stopCriteria){
   for each (particle within S){
      for each (dimension within d){
         first_op = ωv_(i,d)+ φ_p * r_p;
         second_ op = (p_(i,d)- x_(i,d);
         third_ op = φ_g * r_g * (g_d- x_(i,d);
         d[i] = first_op * second_op + third_op;
      }
      Position += d[i];
      if( f(xi) < f(pi) ){
         bestParticleLocalPosition = xi;
         if( f(pi) < f(g) ){
            bestGlobalPosition = pi;
         }
      }
   }
return bestGlobalPosition;
}
```

Application to the smart cities realm and potential results

As surely the reader can imagine after studying this paper, the wide spectrum of applications that can be extracted from Natural Computing and applied to the smart city is, almost solely, bounded by wit limits. When it comes to a city´s realm, the citizens can be seen as *particles*, thus they conform a *swarm*, atomic work element of the Natural Computing paradigm.

On the one hand, the efforts under the current investigation are being driven into the ACO Algorithm spectrum: Even efficient, ant pheromone is simple, primitive; it only marks the shortest path to the livelihood source, but; what if this pheromone concept is *extended* to a *super-pheromone*? A super-pheromone will store dissociated data of a person (i.e.: age, gender, education level, etc), thus, it will be possible to know which *person profile* is transiting for each UPTS section by seeing the user as an ant. More concretely, by applying the schema shown in [*Figure 2. Overview of the currently under development system*] under the epigraph [3. Topology of the system], the UPTS will acquire conscience, knowing who is circulating where, and consequently showing publicity screens according the relevant information for the public. (For instance, it will be more effective to show the publicity related to a new videogame near an institute area when the train is crowded by young people, while a new credit card with certain bonuses will be more appropriate in the UPTS section beneath the financial area of the city.)

On the other hand, GA paradigm is being used as a way to *evolve* a route instead of a chromosome population: by means of a Smartphone application, users will be able to quickly know the best route between two points in the UPTS, as well as backup routes in case of systems breakdown. PSO can be used for studying the data retrieved in the Operations and Control Center (OCC). This will make possible to optimize the system by applying a statistical investigation over the data, detecting statistical outliers and acting in consequence. It is remarkable that the investigation regarding this slope and other Natural Computing paradigms is in an early stage, thus new applications are susceptible to emerge.

Conclusions

In this paper, a newfangled scheme for endowing intelligence to a city UPTS is given, chasing the transition of the city to a *smart city*. In this approach, Natural Computing paradigm will be applied to the system, after a deep investigation that aims to improve the involved paradigms, if possible. Despite the investigation is still being in an early stage, the system is likely to improve the data gathering related to the UPTS in a mastodontic way, allowing the pertinent authorities to improve the system and even monetize the information gathered by the system under development. Moreover, users will be able to enjoy a better use of UPTS, knowing alternative routes in case of systems breakdown and being able to travel in an efficient way.

Acknowledgements

We would like to thank Technical University of Madrid (Universidad Politécnica de Madrid – https://www.upm.es) for the granted permission in order to use the Center of Calculation and Communications for the simulations involved within the realm of this project.

Bibliography

[Blickle, 1996] Blickle, T, Thiele, L. (1996). A Comparison of Selection Schemes Used in Evolutionary Algorithms. Evolutionary Computation 4 (4): 361–394, 1996

[CEOE, 2015] Confederación Española de Organizaciones Empresariales (CEOE). Acciones prioritarias para el desarrollo de las Smart Cities en España (Prioritary actions for the development of the Smart Cities in Spain). Smart Cities Committee, 4-8, Madrid, 2015.

[Darwin, 1859] Darwin, C. On the Origin of Species, London, 1859.

[Dorigo, 1992] Dorigo, M, Di Caro, G. The Ant Colony Optimization Meta-Heuristic, 1999.

[Grenander, 1990] Grenander, L. Heppner, C. *A stochastic nonlinear model for coordinated bird flocks*, 1990.

[Holland, 1975] Holland, J. *Adaptation in natural and artificial systems*. Cambridge, Mass.: MIT Press, 1975.

[Hollands, 2008] Will the real smart city please stand up?. City, [online] 12(3), pp.303-320. Available at: http://www.tandfonline.com/doi/abs/10.1080/13604810802479126#.Vw01M_mLSHs [Accessed 2 Apr. 2016].

[Mendel, 1865] Mendel, G. *Experiments in plant hybridisation*. Cambridge: Harvard University Press, 1865.

[MTA, 2012] Metropolitan Transportation Authority (MTA). Annual Subway Ridership [online]. Available at: http://web.mta.info/nyct/facts/ffsubway.htm[Accessed 5 Apr. 2016].

[Reynolds, 1987] Reynolds, H. *Flocks, herds and schools: a distributed behavioral model*, 1987.

[Rozenberg, 2012] Rozenberg, G., Bäck, T. and Kok, J. Handbook of natural computing. Berlin: Springer, 2012.

Authors' Information

Clemencio Morales *– Computer Engineer, Master on Web Engineering, PhD Candidate at Technical University of Madrid (Spain), Madrid, Spain; e-mail: mail@clemenciomorales.com*

Major Fields of Scientific Research: Natural Computing, Ant Colony Optimization, Smart Cities, Software Engineering.

Luis Fernando de Mingo *– Professor and investigator at Technical University of Madrid (Spain), Madrid, Spain; e-mail: fernando.demingo@upm.es*

Major Fields of Scientific Research: Natural Computing, Software Engineering, Ubiquitous Programming.