Enhancing the Urban Road Traffic with Swarm Intelligence: A Case Study of Córdoba City Downtown

José García-Nieto[1], Enrique Alba[1], Ana Carolina Olivera[2]
jnieto@lcc.uma.es

University of Málaga (Spain)[1]
University Nacional del Sur (Argentina)[2]

UNIVERSIDAD DE MÁLAGA

LENGUAJES Y CIENCIAS DE LA COMPUTACIÓN
UNIVERSIDAD DE MÁLAGA

GISUM

Overview

1. Introduction
2. The Problem: Optimal Cycle Programs of Traffic Lights
3. Optimization Strategy
4. Experimental Framework
    1. Scenario Instance: Córdoba
    2. Parameter Settings
5. Results and Analyses
    1. Performance Analysis
    2. Analysis of Obtained Cycle Programs
6. Conclusions and future work

Introduction

- Nowadays, the excessive vehicular traffic in current cities provokes severe problems related to: pollution, congestion, security, noise, and many others

- Improving the flow of vehicles is a mandatory task

- Traffic lights are configurable devices that partially control the flow of vehicles

- Nevertheless, the increasing number of traffic lights require a highly complex scheduling

- A great number of combinations (color states and phase durations) appear that should be considered (explored) by experts

Motivation:

Providing the experts with **automatic intelligent tools** to obtain optimized traffic lights schedules for large urban areas

Introduction

<u>Hypothesis:</u>
Metaheuristics approaches (concretely PSO) **can find** successful schedules of traffic lights for **heterogeneous urban scenarios** in **reasonable time**

- Our proposal: A Swarm Intelligence approach (Particle Swarm Optimization) coupled with SUMO (Microscopic Simulator of Urban Mobility), to **automatically search quasi-optimal solutions** (traffic lights schedules)

- Case Study: realistic metropolitan area in the city center of Córdoba

The Problem: Optimal Cycle Programs of TLs

- All traffic lights located in the same intersection are governed by a common program, e. g., the combination of color states during a cycle period is kept valid (it must follow specific traffic rules of intersections)

- Main objective: find optimized cycle programs (CP) for all the TLs in a given area. CPs are referred to the time span (or phase duration) a set of TLs, in a given intersection, keep their color states

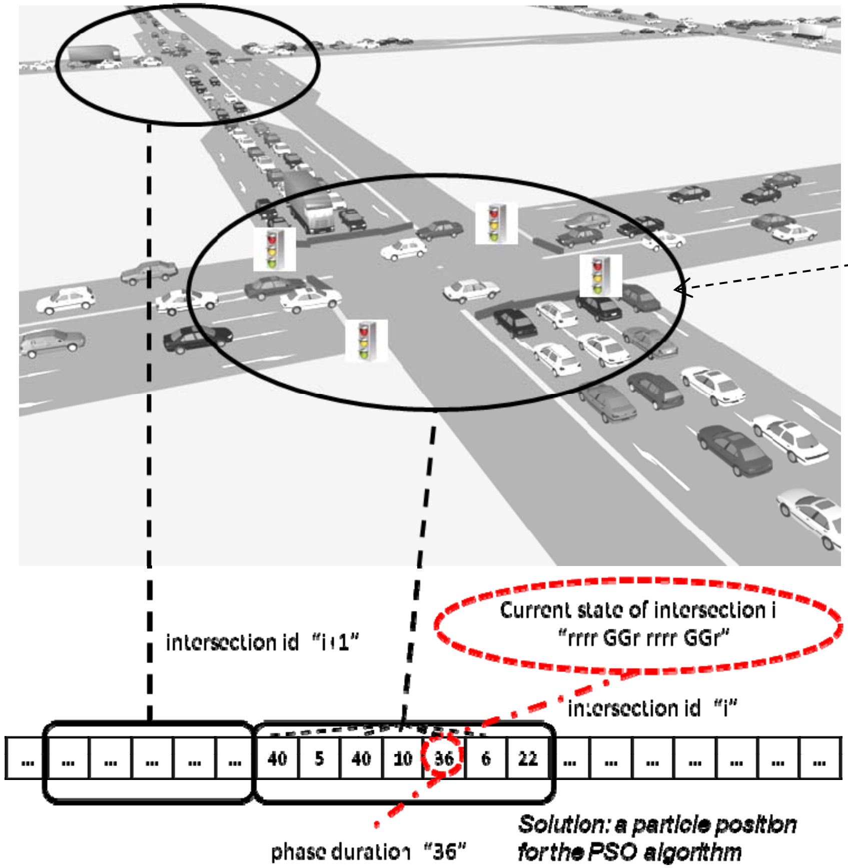
8 TLs with a common CP in intersection:

Paseo de la Victoria/Ronda de los Tejares

- As in real schedules, CPs are designed for established time periods with certain vehicle densities and speeds (rush hours, nocturne periods, etc.)

The Problem: Optimal Cycle Programs of TLs

- Solution encoding: vector of integers where each element represents a phase duration of one state of TLs in intersections (SUMO structure of CPs)



- Adjacent intersections have to be also coordinated in order to improve the global flow of vehicles

The Problem: Optimal Cycle Programs of TLs

- Fitness function: maximizing the number of vehicles that reach their destinations and minimizing the global trip time of all the vehicles, during the simulation time

$$fitness = \frac{TT + SW + (NV * ST)}{V^2 + P}$$

Values collected during the simulation process

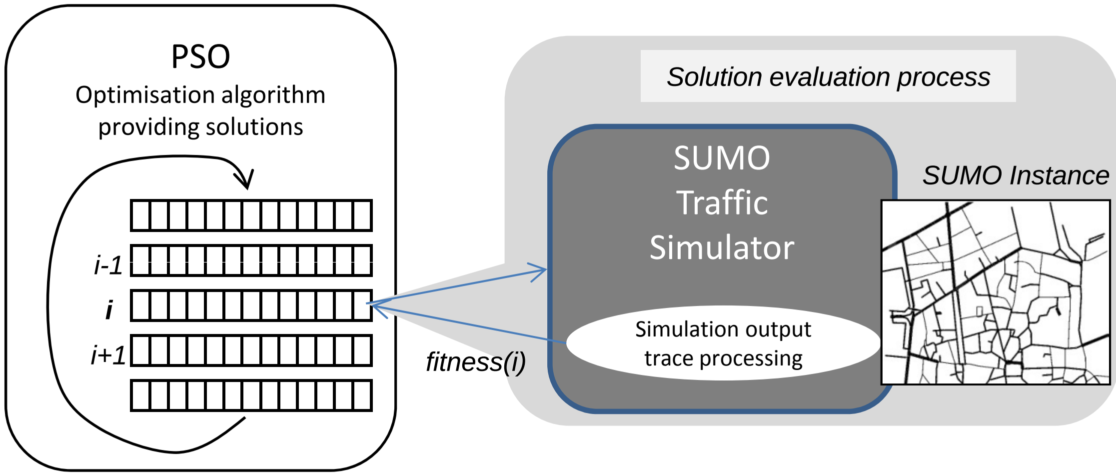
V : Number of vehicles that reach their destinations
TT: Global trip time of all the vehicles
ST: Simulation time
NV: Vehicles that do not reach their destinations
SW: Mean time each vehicle must stop and wait
P: Proportion of colors in each phase and intersection

$$P = \sum_{k=0}^{tl} \sum_{j=0}^{ph} s_{k,j} * \left(\frac{G_{k,j}}{r_{k,j}} \right)$$

tl : Intersection
ph: phase
G: Number of traffic lights in green
r: Number of traffic lights in red
S: phase duration

Optimisation Strategy

- Optimization algorithm (PSO) with Simulation procedure (SUMO)



Algorithm 1 Pseudocode of Standard PSO 2007 for OCP

```
initializeSwarm()
while g < maxIterations do
    for each particle x_g^i do
        b_g^n=bestNeighbourSelection(x_g^i, n)
        v_{g+1}^i=updateVelocity(w, v_g^i, x_g, φ_1, p_g, φ_2, b_g^n)   //Eq. 4
        x_{g+1}^i=Q(updatePosition(x_g^i, v_{g+1}^i))   //Eqs. 3 and 5
        evaluate(x_{g+1}^i)  //SUMO Simulation and Eq. 1
        p_{g+1}^i=update(p_g^i)
    end for
end while
```

Mid-Thread quantisation

$$Q(x) = \Delta \cdot \lfloor x/\Delta + 0.5 \rfloor$$

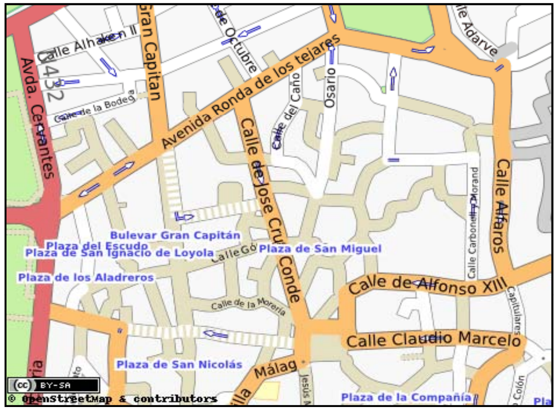
Scenario Instance: Córdoba

- Scenario generated from actual information in real digital maps
- A urban area of approximately 0.75km^2 comprising: Ronda de los Tejares, Alfaros, Claudio Marcero, and Cervantes street
- 30 intersections each one of them including from 4 to 16 TLs. A total number of 152 TLs (solution dimension). Simulation time 500 s
- Three scenario versions with traffic densities: 100, 300, and 500 vehicles

Córdoba Ronda de los Tejares Avenue Spain

Google Map view

OpenStreetMap view



SUMO capture view

Experimental Setup

- Comparison of Standard PSO 2007, Differential Evolution (DE), Random Search (RAND), and SCPG (deterministic cycle program generator provided by SUMO according to human expert information)

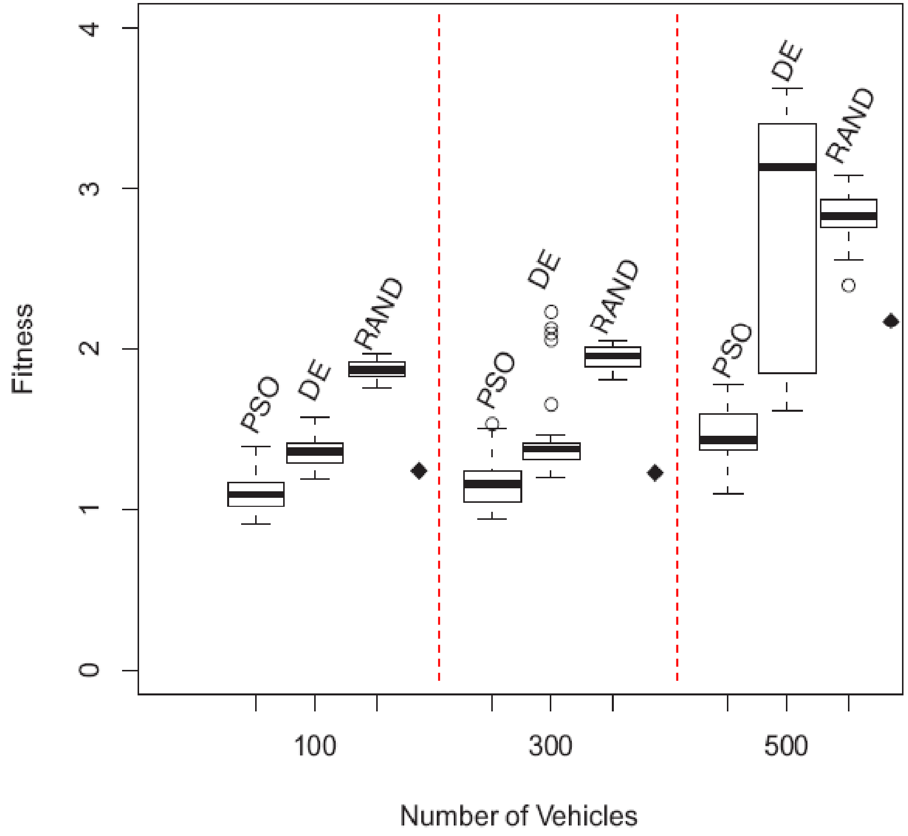
Implementation in C++ MALLBA Library [online available]

- Standard PSO and DE, with populations of 100 vector solutions and performing 200 iteration steps, e. g. total number of 20.000 function evaluations (simulations)
- Random Search also performing 20.000 function evaluations
- Phase durations (variables) initialized in the range of [6,31]

Solver	Parameter	Value
PSO	Swarm Size	100
	Particle Size (N. Traffic Lights)	152
	Local and Social Coefficients ($\varphi_1 = \varphi_2$)	2.05
	Neighborhood size (n)	3
	Inertia Weight (w)	0.7213
DE	Population Size	100
	Individual Size (N. Traffic Lights)	152
	Mutation Constant (F)	0.5
	Crossover Probability (Cr)	0.9

Performance Comparison

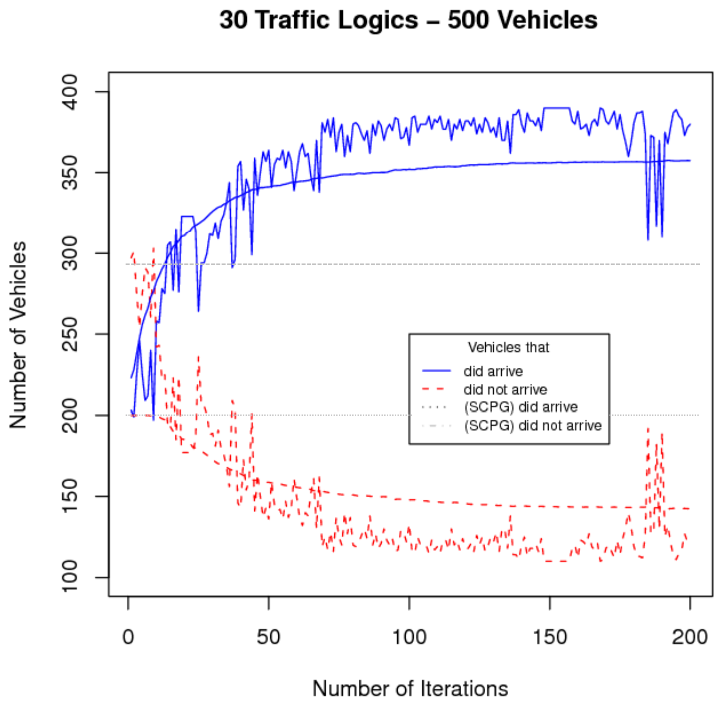
- PSO obtains the best results in general, followed by DE, SCPG (♦) and Random Search
- Statistically, each distribution pair (Wilcoxon) obtained significant differences (α=0.05), excepting for DE and RAND with (500 vehicles)
- The higher the traffic density, the greater the benefits of using PSO

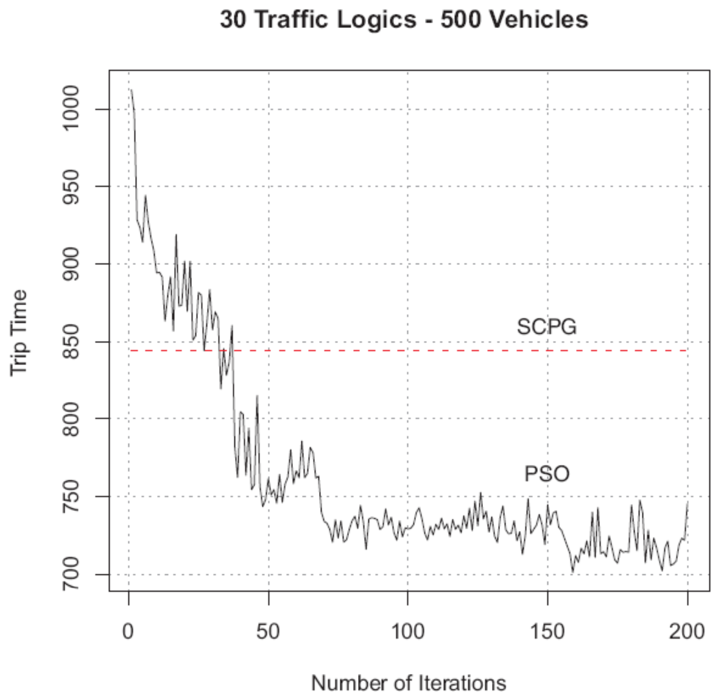


Performance Comparison

- The global trip time becomes shorter as the PSO approaches the stop condition (improvement of 17.45% respect to SCPG solutions)

- The number of vehicles that reach their destinations increases along with the search progress

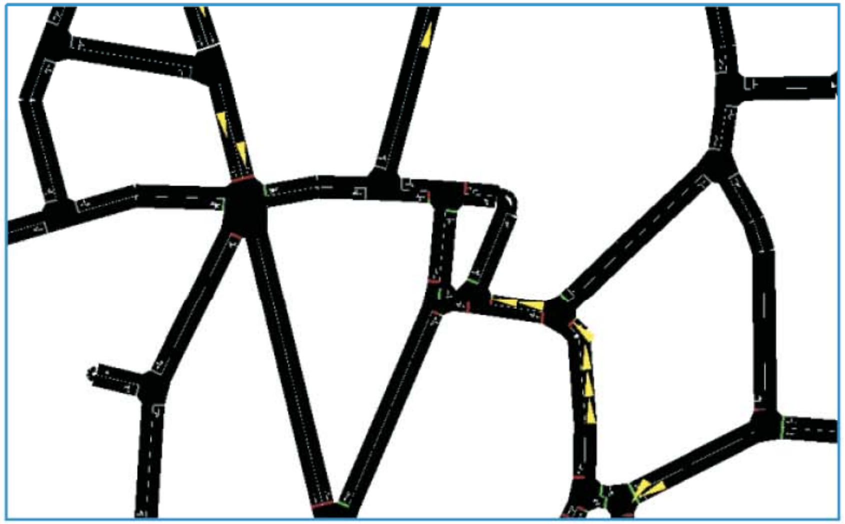




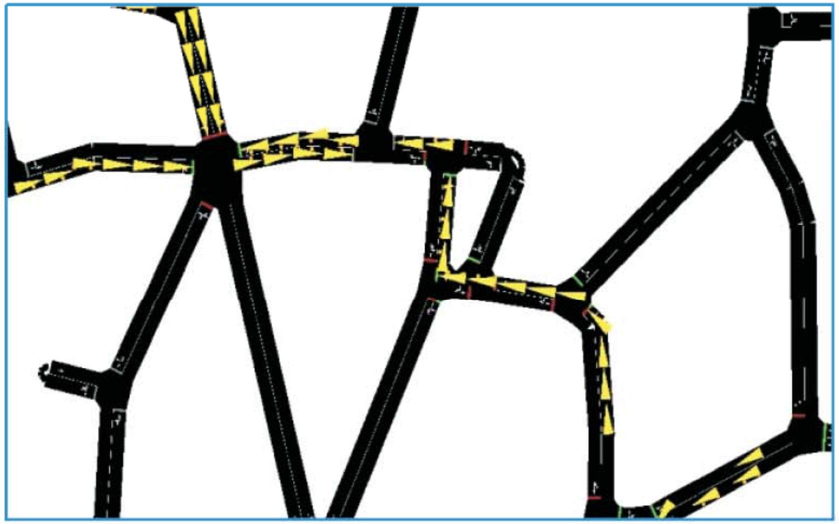
Performance Comparison

- Simulation snapshots of resulted cycle programs generated by PSO and SCPG with 500 vehicles (initial traffic density)

- A low traffic density can be observed in PSOs' solutions, but traffic jams appeared in SCPG ones



PSO



SCPG

Conclusions

- We have proposed a Swarm Intelligent approach that, coupled with the SUMO traffic simulator, is able to find successful cycle programs of traffic lights. In concrete, we have focused on a metropolitan area of the of Córdoba city
- After the experimentation, we test our initial hypothesis:
  - Our proposal performed efficiently for the studied instance. In comparison with DE, Random Search, and SCPG, our PSO showed the best performance
  - PSO scales adequately in terms of traffic density with: 100, 300, and 500
  - Obtained CPs by PSO can improve both, the global trip time and the number of vehicles that reach their destinations
- Future work:
  - Tackling the problem with other metaheuristics
  - New large instances as close as possible to real scenarios of a whole city

Thank you so much!!