COMPARISON BETWEEN GENETIC ALGORITHM AND PARTICLE SWARM OPTIMIZATION FOR POSITIONING A SET OF CAMERAS FOR VIDEO SURVEILLANCE

David Strubel [1;2] Mark Bastourous[1] Olivier Morel[1] Naufal M. Saad [2] David Fofi[1]

Abstract— **This paper have to aims of optimizing a vast coverage area to allow an image reconstruction using mosaicing techniques. Among the investigated methods to find the best camera positions, two of them are studied, namely the Particle Swarm Optimization (PSO) and the Genetic Algorithms (GA). After having performed many experiments to compare the algortithms, the GA is chosen for this performance and this adaptability to the problem. To validate the proposed method, it is applied on area of irregular shape and with the cameras mounted on the Unmanned Aerial Vehicles (UAVs). V-REP [14] is used to simulate the UAVs in the environment (indoor or outdoor). The simulation validates the efficiency of the proposed method to find the number and the poses of the cameras. Then by using the images acquired it is possible to compute mosaic images and control the area.**

Keyword: Coverage, Genetic Algorithm, Optimization, Mosaic, UAV, Application

I. INTRODUCTION

The optimal positioning of a cameras network is tricky and no efficient solution exists to solve it in a complex environment. The aim of this work is to provide a flexible and tunable solution for this problem and to analyse the performance of the current state-of-the-art techniques. The final objective of our research is to design a global optimization scheme allowing a camera network to self-organize and self-reconfigure, according to set of fixed priorities and constraints, in order to ensure a maximum coverage. Self-organization have to be performed to be efficient in realistic conditions. Within this context, it is important to assess the actual performance and limits of the state-of-the-art algorithms. First, we study the following contribution study and compare three standard algorithms, namely the Random Selection (RS), the Particle Swarm Optimization (PSO) and Genetic Algorithm (GA), in terms of efficiency and quality. The quality is evaluated with the coverage rate and the efficiency by the numbers of iteration for the optimisation to converge.
In addition the solution proposed is supported by an adapted and optimized cost function, specially adjust to the UAV problems.
We propose :

- A comparison between different algorithm from literature and the same family.
- Using the Genetic Algorithm for position many cameras in a big area.
- A cost function adapted to the problem of coverage using UAV.

1 LE2I UMR6306, CNRS, ENSAM, Univ. Bourgogne Franche-Comte
2 CISIR, Universiti Teknologi Petronas

II. RELATED WORKS

Sensor positioning problem has been investigated since in the last decades, mainly for video surveillance [1]. Without any additional constraint, this problem is a NP-Hard as stated in [1]–[5] Two solutions have been proposed to optimize the coverage of the area with a sensors network. The first one is based in the Art Galleries Problem (AGP) [1], [2]. AGP the position of body guard is optimized in a museum. The second is based on the Wireless Sensors Networks [6]–[9] trying to find the best position to design an efficient network which can collect data with any kind of sensors. However, the solution proposed to this problem works only with some constraints such as the sensor has 360 field of view and no obstacle. One of the algorithm commonly used is PSO as detailed in [10], [11]. In Zhou et al [10], some experimental results are provided and one solution running in real time is proposed. However, the scene used for the experiments is rather small and many cameras are employed to fully cover it. On the other hand, Reddy et al [11] optimize the position of the cameras by using a cost function but also handling resolution and lighting. The multi-goal approach affect the final solution. In Boeringer et al [11] also introduces the concept of acceptable response, allowing non-optimal/sub-optimal solutions. If the coverage score is good enough, the solution is accepted and not locked by the research of an optimal solution. Our paper is based on [10], [11], attempting to extend it by testing GA and PSO on different environment (basic room, big room, non-square shape) as well as for drone positioning.

III. OPTIMISATION OF CAMERAS POSITIONING

A. Objectives

The main purpose of our work is to estimate the position of n cameras surveying a given area in order to maximize the visual coverage. The cameras are mounted on a UAV and that each camera is assume to provide a top view of the area. Each cameras coverage is defined by the projection of the visual field onto the ground. Accordingly the combination of all the acquired images from the cameras allows to build a mosaic of the area to be checked. In order to assess the best

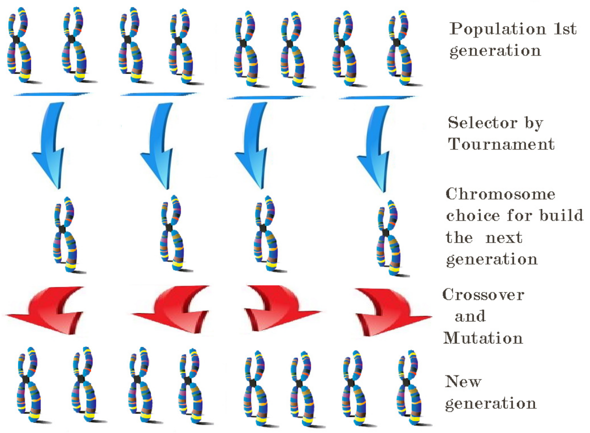


Fig. 1. GA explanation, from a generation to another.

algorithm among PSO and GA have been investigated. The following subsections will give an overview of both.

B. Particle Swarm Optimization

PSO is an evolutionary method [17], which aims at optimizing a problem by iteratively improving a candidate called particle. The quality of a particle is evaluated by a cost function. Thereby, the best particles are selected for the next generation. A new set of particles is defined by randomly moving around the best solution given by the previous iteration. The search-space around the best solution is controlled by a parameter called inertia. Repeating this method at every iteration push the algorithm to converge. To use properly the PSO few parameters need to be defined:

- The number of particles for each iterations.
- How the particles are initialize.
- The value of inertia.
- The cost function.

After several test on the different environments described in section IV.B the PSO are configured with an inertia of 0.5, the number of particle are 100, and the initialization of the paricles are fully random. Moreover the cost function used for the PSO is defined in section III.D

C. Genetic Algorithm

Motivated by Darwin′s theory of evolution and the concept of natural selection, the GA use processes analogous to genetic recombination and mutation. To promote the evolution of a population to reach a predefined goal [12], [18] . Such kind of algorithms require the definition of a genetic representation of the problem and the cost function is used to evaluate the solution. The candidate solution is represented by a data structure named chromosome, which is the equivalent of a particle for the PSO. The cost function and the data structure is the same for all the algorithms. Mostly it is to compare properly the algorithms.

The genetic algorithm work as iterative process. Every iteration are called generation and the chromosome of the actual generation are the offspering of the previous generation (see Fig.1. To pass from an generation to the other a few steps are necessary (see Figure 1):

1. For each sub-set, a selection is made, in order to select the most attractive chromosome (i.e. the one that maximizes the cost function).
2. Basic operations as cross-over and mutation are performed, on the selected chromosome to give rise to the next generation. [15]
3. The process is repeated until the convergence point or the generation boundary are reached. GAs allow more flexibility than PSOs however require more parameters to be configured.

In our experiments, we fixed the number of chromosomes to be 90, the operator used are the mutation and cross over with the mutation rate to be 0.001 and the crossover rate to be 0.919. The initial population are fully random and the selection is done by using a tournament selection method [16].

D. Cost function

Since the goal is to maximize the visual coverage of the camera network. They are expressed as a list of the cameras parameters as in the equation (1)

$$Vs = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix}_{1 \leq i \leq n} \tag{1}$$

Where n is the number of cameras.
$x; y; z$ are the position of the camera in the Cartesian coordinate system.
The x and y are limited to the boundary fixed depending the area:

$$\begin{matrix} 0 \leq x \leq \text{width area} \\ 0 \leq y \leq \text{heigth area} \end{matrix} \; x, y \in \mathbb{N} \tag{2}$$

The z are chosen in a list of possible altitude predefined.

Thereafter the cost function is designed to qualify the solution, as follows:

$$\sum_{i=1}^{n} = \frac{cover(i)}{size(grid)}_{1 \leq i \leq n} \tag{3}$$

Where n is the number of cameras; $Grid$ represents the discretization of the ground plane (floor);
$Cover()$ is a function which computes the area on the ground which is covered by at least one camera;
$Size()$ is the dimension of the full area which must be covered.
Camera projection model is not explicitly taken into account, but the ground-projected visual field instead, as described in Figure 2. The equation (3) of the cost function can be also summarize in a basic pseudo-code in Algortihm : Coverage computation.

IV. Experimentation

A. Context of experimentation

In order to compare the two algorithms and evaluate their performances, many test on different scenarios depicted in

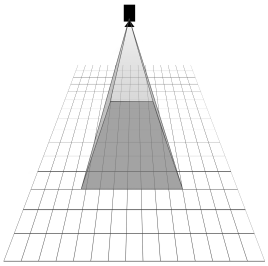


Fig. 2. Projection of the camera on the ground.

Algorithm . Coverage computation

procedure COST FUNCTION($Cameras$)
  $covered = 0$
  while $i = 1$; $i >$ size of the Grid **do**
    while $j = 1$; $j >$ number of cameras **do**
      if $Grid_i$ is visible by $Camera_j$ **then**
        $covered = Grid_i$
  return ...

Figure 3, with have different sizes and shapes of rooms, where:

- z is the height of the camera between (within the range [1z ;z].
- Figure 3.a is an area of size 12080 (named Room).
- Figure 3.b is an area of size 240160 (named Big room)
- Figure 3.c is an area of size 12080 (named Room U)
- Figure 3.d is an area of size 12080 (named Room L)
- Figure 3.e is an area of size 24080 (named Big room L)

z=1		GA		PSO		RS	
		GT	NC	GT	NC	GT	NC
Room	120x80	16	20	16	20	16	20
	240x160	64	70	64	70	64	70
Room U	120x80	12	20	12	20	12	20
z=2		GA		PSO		RS	
Room	120x80	4	10	4	10	4	10
	240x160	16	20	16	20	16	20
Room L	120x80	3	10	3	10	3	10
	240x160	15	20	15	20	15	20

TABLE I

DESIGN OF EXPERIMENT FOR COMPARE THE EFFICIENCY OF RS, PSO AND GA IN DIFFERENT CONDITION. (GT IS GROUND TRUTH AND NC IS NUMBER OF CAMERAS).

The design of experiments (see table 1) is designed to identify the most efficient algorithm for the positioning of a set of cameras. GT that represents the Ground Truth is the minimum number of the cameras required to fully cover a given area. The size of the area has been selected so that the GT can be easily estimated. NC is the maximum number of cameras used for the experiments. For each experiment a solution is computed for a number of cameras from 1

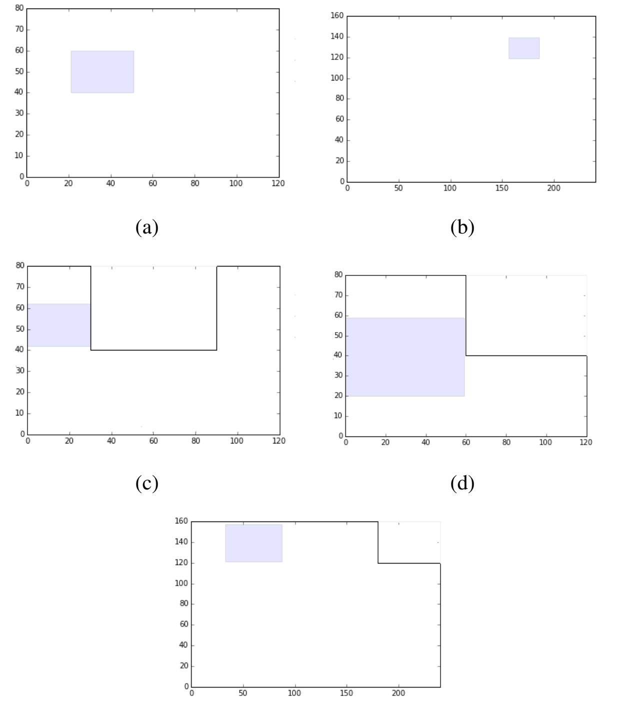


Fig. 3. The scenarios used for the experiments: (a), (b), (c) are with z=1 and (d), (e) with z=2. The grey rectangle represents the field of view of one camera projected onto the ground.

to NC. Note that the PSO and GA are also compared with the Random Selection algorithm(RS), to ensure their performances are better than a pure random process. In order to compare the different algorithms in similar conditions, only 10000 calls of the cost function is allowed for each set of cameras.

B. *Analyse of the results*

After having performed the design of experiment (see table I) it appears the RS is always the worst solution (more obviously on Figure 4 Figure 6), as excepted we compare PSO and GA with RS to check their performance with supposedly the worst possible solution. The GA and PSO algorithm are close but the result was very different according to the parameters of the experiment. In some case the GA is more efficient (example in Figure 5) particularly in the case where the search space is large. Instead PSO is much more effective to optimize small areas (example in Figure 6). This efficiency is explained by the small variety of solution introduced by the PSO. However in the mean time, this small variety is perfect to optimize fastly in a small search space or when the local minima is not too much important. Although the variety introduced by the GA to pass over the local minima. This variety impacts the optimization accuracy and may have more difficulties to find an adjusted position. PSO is efficient when the area of the room is reduced whereas the GA is efficient for larger rooms. From the results obtained by both algorithms a simulation has been performed using

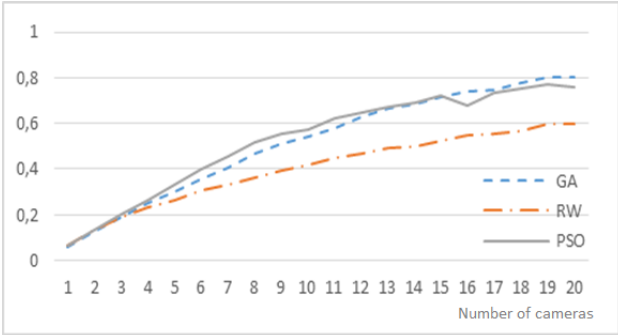


Fig. 4. Comparison of GA, RS, PSO algorithms with a Z between [1/2; 2], in the big room with L shape 240x160 and ground truth equal to 15.

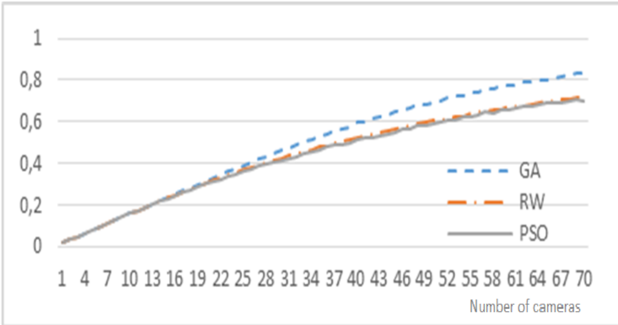


Fig. 5. Comparison of GA, RS, PSO algorithms with a Z equal to 1, in the big room 240x160 and ground truth equal to 64.

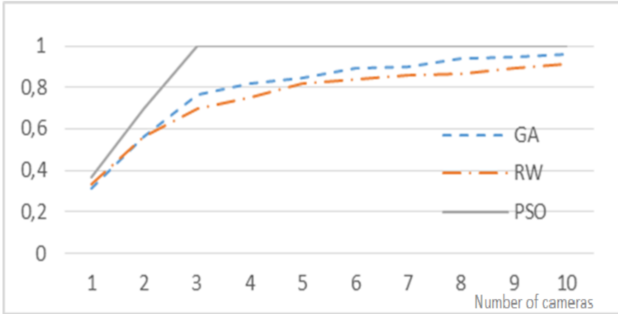


Fig. 6. Comparison of GA, RS, PSO algorithms with a Z between [1/2; 2], in the room with L shape 120x80 and ground truth equal to 15.

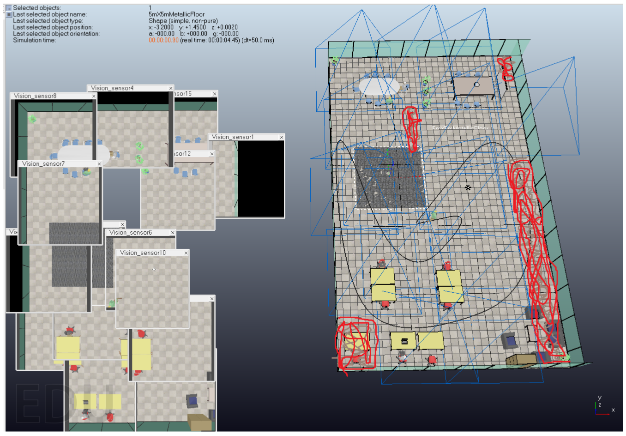


Fig. 7. Demonstration of the simulation with a robot simulation. This demonstration is with a set of 15 cameras and at the left side the mosaicking of the cameras is visible.

the robotics simulation tool VREP as presented Figure 7.

V. Application and Other Results

Thanks to the preliminaries studies the GA seems more suitable to optimize the coverage problem. Based on this finding other experiment was made in other sizes and complexity of environments. the optimisation and the coverage rate instead of complex area stay similar to the previous experiment which are did it with stochastic algorithm as GA and PSO. In addition these algorithm does not use heuristic, so their behavior stay the same and are not depending to the area to cover. The following experiment are done in mush bigger search space and a more complex area although the GA was used with the same parameter excepted the number of generation was free, the GA stop when the convergence point are reached.

A. Indoor coverage

The indoor experiment validate the efficiency of the GA for a same search space then the precedent section with a bit bigger amount of cameras although the area to cover is much more complex as in the Figure 8.

B. Outdoor coverage

The outdoor experiment helping to show the efficiency of GA in a bigger area and in a realistic environment. In the Figure 9.a the area to control form GPS images. From the GPS images the shape of the area to cover are extracted (see Figure 9.b) and the GA are performed to optimize the position of the cameras network the result are in visible in Figure 9.c.

VI. Conclusion

In this paper, the problem of coverage for video surveillance was formalized as an optimization problem. Genetic Algorithms (GA) and Particle Swarm Optimization (PSO) were investigated and compared. Each algorithms has its own drawbacks and advantages:
PSO is more efficient in small environments. Indeed, when the inertia is relatively small, then from an iteration to the other, the algorithm evolves in a close neighborhood. While in contrast, when the inertia is bigger, the algorithm tends to behave like a random selection.
GA is more generic and more efficient, but its parameters have to be adapted to the specific configuration. The future work will be focus on continuing to optimize the coverage and to adapt the solution in the robotic context and application. The optimization of the coverage could take the direction to hybridate GA and PSO in order to combine the advantage of both algorithms or to continue only with the GA but with an adapted and with dynamic parameters. In the other direction the actual result may be extend to the problem of coverage path planing, with a Unmanned Aerial Vehicle to do the control of the area. Also this extension will be linked with the mosaicking application.

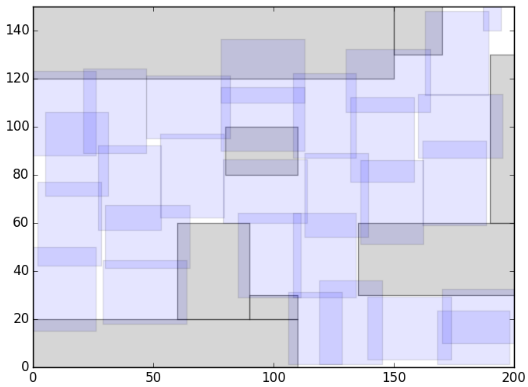
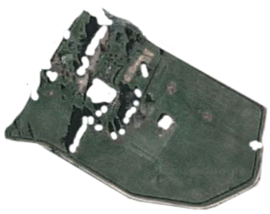


Fig. 8. Cameras positioning with a set of 29 cameras to cover 97.92% of the area.



(a) GPS image of the area to area

(b) Mask of the area to cover

(c) Area cover by the network of cameras

Fig. 9. Coverage area from satelit image with 35 cameras for 86.3% of coverage

References

[1] Chin, W. Ntafos, S.: Optimum watchman routes. Inform. Process. Lett. (1988)

[2] Moeini, M., Krller, A., Schmidt, C. Une Nouvelle Approche Pour la Resolution du Probleme de la Galerie dArt.

[3] Erdem, U. M., & Sclaroff, S. (2006). Automated camera layout to satisfy task-specific and floor plan-specific coverage requirements. Computer Vision and Image Understanding, 103(3), 156-169.

[4] Packer, E. (2008). Computing multiple watchman routes. In Experimental Algorithms (pp. 114-128). Springer Berlin Heidelberg.

[5] Zhao, J., Cheung, S. C., & Nguyen, T. (2008). Optimal camera network configurations for visual tagging. Selected Topics in Signal Processing, IEEE Journal of, 2(4), 464-479.

[6] Song, B., Soto, C., Roy-Chowdhury, A. K., & Farrell, J. (2008, September). Decentralized camera network control using game theory. In Distributed Smart Cameras, 2008. ICDSC 2008. Second ACM/IEEE International Conference on (pp. 1-8). IEEE.

[7] Liu, L., Zhang, X., & Ma, H. (2010). Optimal node selection for target localization in wireless camera sensor networks. Vehicular Technology, IEEE Transactions on, 59(7), 3562-3576.

[8] Ma, H., Yang, M., Li, D., Hong, Y., & Chen, W. (2012, March). Minimum camera barrier coverage in wireless camera sensor networks. In INFOCOM, 2012 Proceedings IEEE (pp. 217-225). IEEE.

[9] Wang, Q., Wu, J., & Long, C. (2013, October). On-line configuration of large scale surveillance networks using mobile smart camera. In Distributed Smart Cameras (ICDSC), 2013 Seventh International Conference on (pp. 1-6). IEEE. Transactions on, 2004, vol. 52, no 3, p. 771-779.

[10] Zhou, P., & Long, C. (2011, October). Optimal coverage of camera networks using PSO algorithm. In Image and Signal Processing (CISP), 2011 4th International Congress on (Vol. 4, pp. 2084-2088). IEEE.

[11] Reddy, K. K., & Conci, N. (2012, October). Camera positioning for global and local coverage optimization. In Distributed Smart Cameras (ICDSC), 2012 Sixth International Conference on (pp. 1-6). IEEE.

[12] Boeringer, Daniel W. et Werner, Douglas H. Particle swarm optimization versus genetic algorithms for phased array synthesis. Antennas and Propagation, IEEE

[13] [13] Shi, X. H., Liang, Y. C., Lee, H. P., et al. An improved GA and a novel PSO-GA-based hybrid algorithm. Information Processing Letters, 2005, vol. 93, no 5, p. 255-261.

[14] http://www.coppeliarobotics.com/

[15] SRINIVAS, Mandavilli et PATNAIK, Lalit M. Adaptive probabilities of crossover and mutation in genetic algorithms. Systems, Man and Cybernetics, IEEE Transactions on, 1994, vol. 24, no 4, p. 656-667.

[16] MILLER, Brad L. et GOLDBERG, David E. Genetic algorithms, tournament selection, and the effects of noise. Complex Systems, 1995, vol. 9, no 3, p. 193-212.

[17] EBERHART, Russ C., KENNEDY, James, et al. A new optimizer using particle swarm theory. In : Proceedings of the sixth international symposium on micro machine and human science. 1995. p. 39-43.

[18] HOLLAND, John H. Outline for a logical theory of adaptive systems. Journal of the ACM (JACM), 1962, vol. 9, no 3, p. 297-314.