Multi Agent Path Planning Strategies based on Kalman Filter for Surveillance Missions

Desirée Gentilini, Nicola Farina[1], Enrico Franco[2], Anna Elena Tirri, Domenico Accardo, Rosario Schiano Lo Moriello[3], Leopoldo Angrisani[4]

***Abstract*—The paper is based on the definition of a path planning strategy for surveillance missions with a system of multiple Unmanned Aircraft by means of a Kalman Filter technique. The developed method aims at finding a set of commands for the network of aircraft able to minimize a cost function whose definition depends on the mission. The approach adopted in this paper comprises several steps. The first one is based on the development of a target tracking algorithm to provide information on both target and drones motion on the surveillance area by means of a Kalman Filter and Bayesian network. Then, the objective functions can be defined depending on the relative position between aircraft and target. Finally, a heuristic approach allows finding the set of commands for the aircraft deployment over the surveillance area that maximize the utility function during the mission. The results demonstrate the ability of the tracking algorithm to provide accurate estimate of the target motion and the good capability of the whole system to react to the Command centre inputs based on the defined utility functions and decision making strategy.**

***Index Terms*—Path Planning, Kalman Filter, Target Tracking, Surveillance**

I. Introduction

MULTI agent systems have increased their level of importance in the research community over the years due to the possibility to employ these systems for dual-use applications [1-3]. In fact, they are adopted in several applications in both military and civilian domain, in industrial and personal applications. Multi agent systems are often adopted to accomplish tasks where cooperation between different robots is preferred, like situations that are hazardous or time and power consuming for a single agent. The functions that the "network" of robots can execute range from target tracking to dynamic coverage, from autonomous navigation to collision avoidance and surveillance [4, 5].

[1] D. Gentilini and N. Farina are with Pangea Formazione, Via Gaspare Gozzi 55, 00145 Rome, Italy (desiree.gentilini@pangeaformazione.it and nicola.farina@pangeaformazione.it).

[2] E. Franco is with Pangea Formazione, Via Gaspare Gozzi 55, 00145 Rome, Italy and INFN, Sezione di Roma, P.le A. Moro 2, I-00185 Rome, Italy (enrico.franco@roma1.infn.it).

[3] A.E. Tirri, D. Accardo, And R.S. Lo Moriello are with Department of Industrial Engineering, University of Naples Federico II, P.le Tecchio 80, 80125 Napoli, Italy (annaelena.tirri@unina.it, domenico.accardo@unina.it, and rschiano@unina.it).

[4] L. Angeisani is with Department of Electric Engineering and Information Technology, University of Naples Federico II, Via Claudio 29, 80125 Napoli, Italy (angrisan@unina.it).

The adoption of teams of multiple unmanned systems permits to share information about situational awareness and planning strategies and to improve the overall performance of the system in executing a particular mission.

The network of robots can be deployed as a team in different configurations; they can be based on centralized or decentralized control. In the first case, the UAVs may be connected to a control node, such as a human based control station, which receive information, compute a plan of actions and send specific command to each robot [1]. In the other cases, there is a link that connects the robots that permits to exchange mutual information and to formulate their own path planning [2-4]. Some hybrid approaches also exists where the UAVs can share data with other manned and unmanned vehicles [5]. In all cases, the team of robot is coordinated through a mutual exchange of information. As a consequence, the availability of a communication framework is fundamental for the deployment and control of the multi robot system. For instance, controlling the team of UAVs requires an effort that can be similar to the one required to control a single robot, provided that an adequate communication system is available [6].

Usually, the agent adopts wireless communications to exchange information. This requires that a wireless data link with adequate performance is provided to the robots in terms of robustness and reliability. In order to compensate some drawbacks of the communication link specific swarming control strategies can be exploited. The main problems are related to time delays, data loss, quantization, and coding. Therefore, an optimal path planning strategy that takes into account these drawbacks must be identified.

In case of swarms, the path planning problem consists in finding the best sequence of commands to be assigned to each drone in order to gain the highest level of probability to fulfill a specific task, i.e. in order to maximize a specific cost function. In order to define this problem, the following items must be provided:

1) An analytical or numerical model of behavior for each drone. It includes:
    a. A dynamical model that describes how each drone executes each command;
    b. A stochastic model that describes the level of uncertainty that is associated to the execution of each task;
2) A cost function that allows for a quantitative evaluation of the level of fulfillment of the assigned task;
3) A proper decision making strategy that allows for selecting the path that minimizes the above mentioned cost function.

978-1-5090-1131-5/16/$31.00 ©2016 IEEE

The identification of a path planning strategy can be approached drawing inspiration from biology, the so-called bio-inspired or bio-mimetic technologies. In this case, the link with biology is dual because the flying platforms are inspired by birds or insects, and swarming is the way that is adopted by birds and insects to organize their group activities, such as transfers, exploration, and food procurement. For example, Particle Swarm Optimization (PSO) is a popular based optimization technique proposed by Dr. Kennedy and Dr. Earhart in 1995. It is inspired by the social behavior of bird flocking and schooling of fishes [7]. In PSO, each 'bird' or 'fish' act as a particle and the 'flock' or 'school' is the swarm. Ant Colony Optimization (ACO) algorithms have been proposed by M. Dorigo in 1992. It is developed to show the behavior of real ants to provide heuristic solutions for optimization problem. It solves the problem of path planning in mobile robot by using the ability of optimization in the process of ant searching food [8]. The Artificial Bee Colony (ABC) is a population based heuristic algorithm introduced by Dervis Karaboga in 2005 for solving constrained optimization problem [9]. It is inspired by collective behavior of honeybees to find food sources around the hive. The firefly algorithm (FA) is a metaheuristic algorithm developed by the author Xin-She Yang in 2008. It is inspired by the flashing behavior of fireflies. The Firefly Algorithm (FA) is a population-based technique to find the global optimal solution based on swarm intelligence, investigating the foraging behavior of fireflies. The primary purpose for a firefly's flash is to act as a signal system to attract other fireflies [10].

The paper aims at finding a path planning strategy for surveillance missions with a SWARM system. In particular, the purpose of the study is the identification of a set of command for the network of drones able to minimize a cost function defined on the basis of the task at hand. An original approach is presented, involving the adoption of Kalman Filter to support the decision making strategy. This approach is flexible to minimize different cost function depending on specific user requirements.

The paper is organized as follows. In Section II, the target tracking algorithm based on Kalman Filter is presented; it provides information on the drones and target dynamic model and defines the scenario in which the model is applied. Section III identifies a cost function that determines qualitatively the fulfillment of the assigned task. Section IV is devoted to the decision making strategy that allow determining the best path among those available to minimize the defined cost function. The results of the developed model are shown in Section V. In particular, the target tracking algorithm performance is reported and the set of commands for the deployment of the drones are individuated. Section VI reports conclusion and future works.

II. MODEL

In this paper, the surveillance area is modeled as a two-dimensional Cartesian rectangle where a single target can move. Reference time interval is defined by the fixed interval between two successive position measurements by the Command Centre radar.

The model does not take into account the time latency due to signal acquisition and aircrafts operational system. Therefore each time interval is characterized by

1. Target position measurement $(y1_k, y2_k)$
2. Aircraft position revelation $(y1_k^D, y2_k^D)$, D=1..Nd
3. Prediction of future target position
4. Aircrafts control update

A. Target Path Model

Target motion on the surveillance area is modeled as a linear dynamic system with a random acceleration:

$$\dot{x}(t) = \boldsymbol{A}x(t) + B(t) \tag{1}$$

where x(t) is a four-components state vector, accounting for both position (first two components) and velocity(last two). $\boldsymbol{A}$ is a 4x4 transition matrix, and B is a white Gaussian noise with the following properties:

$$E[B(t)] = 0$$
$$E[B(t_1)B^T(t_2)] = \Sigma_B \delta(t_1 - t_2) \tag{2}$$

i.e. it is assumed that B is drawn from a zero mean multivariate normal distribution with covariance matrix Σ_B, and δ is the Dirac delta.

The general solution of eq. (1) can be written as

$$x(t) = e^{A(t-t_0)}x(t_0) + \int_{t_0}^{t} e^{A(t-\tau)}B(\tau)d\tau \tag{3}$$

Introducing the discretization time $t = t0 + k\Delta$, it is possible to express the solution of the above equation in terms of the interval of the available measurements as:

$$x(k) = \boldsymbol{T}\,x(k-1) + \phi(k) \tag{4}$$

where $\boldsymbol{T} = e^{\mathbf{A}\Delta}$, $\phi(k)$ is a white Gaussian noise with $E[\phi(k)] = 0$ and $E[\phi(k)\phi^T(j)] = \boldsymbol{Q}\delta_{ij}$, where

$$\boldsymbol{Q} = \int_0^{\Delta} e^{\boldsymbol{A}\tau}\Sigma_B e^{\boldsymbol{A}^T\tau}\,d\tau \tag{5}$$

is the process covariance matrix.

The position measurement y at time k is affected by a normal white noise M with H covariance, and is related to the state vector through:

$$y_k = \boldsymbol{C}x_k + M(k) \tag{6}$$

where $\boldsymbol{C}$ is a 2x4 matrix (it is assumed that just the position is measured).

The optimal estimate for the state vector is calculated by means of the Kalman filter,

$$\hat{x}_k = x_{k|k-1} + \boldsymbol{K}_k\left(y_k - \boldsymbol{C}x_{k|k-1}\right) \tag{7}$$

where $\mathbf{K}$ is the so-called Kalman gain, y_k is the position measurement at time k and $x_{k|k-1}$ is the best estimate of the state vector at the time k given the full knowledge of it at the k-1 time.

As we have seen, in order to apply the Kalman Filter we had

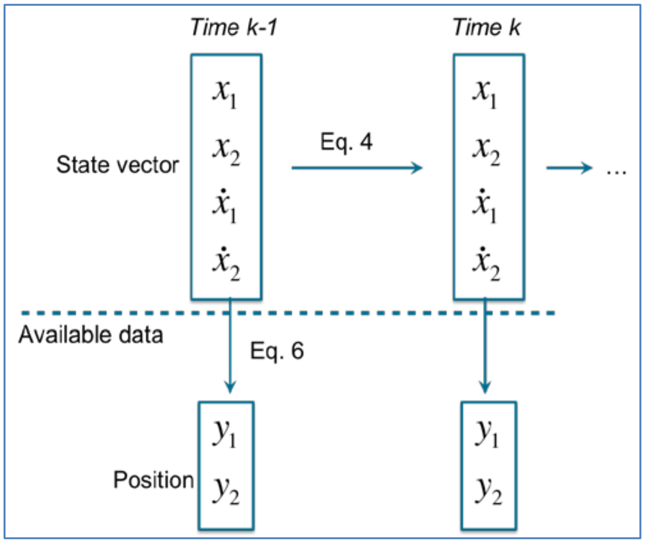

Fig. 1. Bayesian network for the target motion.

to make some assumptions about the nature of the random noises involved. If noises are not normally distributed, or if the dynamic is not linear, an analytical solution of the optimal estimation of the state vector is often unfeasible. However, in general the problem can be modeled as a Bayesian network where the nodes are defined by the state vector and the measurements, as shown in figure 1. Markov Chain Monte Carlo methods, such as the Gibbs Sampling, can be used to draw from the posterior distribution of the state vector given the set of available measurements in a given moment. Therefore, Bayesian networks can handle not only linear dynamic systems but also more general relations among nodes.

We randomly generate target positions according to the equation of motion and measurements (1 and 6). Then, we tried to reconstruct the state vector using both the Kalman Filter and a Bayesian Network (BN) shown in fig.1. We use the JAGS library[11] to draw from the BN and R packages *dlm* [12] and *FKF* [13]to implement the KF. A comparison of the results is shown in fig. 2.

B. Drones Motion

The aircrafts are directed by the command center and this characteristic is reflected in the equation motion by an additional control term respect to the target. Actually for each aircraft the motion equation is:

$$x_k^D = \boldsymbol{E}x_{k-1}^D + \boldsymbol{F}u_{k-1}^D + w_{k-1}^D \quad (8)$$
$$y_k^D = \boldsymbol{G}x_k^D + v_k^D \quad (9)$$

where D is the aircraft index, $\boldsymbol{E}$ is a 4x4 matrix (the respective of the $\boldsymbol{A}$ matrix in the target motion equation) , $\boldsymbol{F}$ is the covariance matrix of the white standard normal Gaussian noise u_{k-1}^D . The last term is the deterministic control input. The measurement equation is analogous to the target's one.

In order to take into account the physical limitations of the aircraft motion, we introduced a viscosity term proportional to the speed in the transition matrix $\boldsymbol{E}$ (while the $\boldsymbol{A}$ matrix in eq. (1) describes a uniform motion on a straight line). Furthermore, we do not allow the input control let the aircraft flight at a speed greater than a threshold value Vmax.

The first of the two modifications above can be fully treated within the Kalman Filter formalism. The second one represents a constraint on the space of the possible control inputs the Command Centre can give to an aircraft.

III. Objective Function Definition

In order to define a collaborative strategy of aircrafts deployment, we have defined three different Utility functions aimed to fulfill different tasks. The Utility function evaluates the 'goodness' of the aircrafts positions with respect to the target.

The first Utility function is maximized when the distance target-aircraft $d(x, x^D)$ is minimized and depends only on the *closest* aircraft

$$U\big(x(t), x^D(t)\big) = F(\min(d(x, x^D))) \quad (10)$$

where F is a strictly decreasing function.

The above function is useful when the task requires that at least one aircraft is as close as possible to the target.

In some circumstances, it is not necessary to reach the target to fulfill a specific task, but a proper distance can be established and defined by the command center.

Therefore we implemented a second Utility function where, besides the high proximity to the target, the position of the aircraft is rewarded with respect to specific distances from the target r_1, r_2

$$U\big(x(t), x^D(t)\big) = F(d(x, x^D); n_1, n_2; r_1, r_2) \quad (11)$$

where n_1 is the number of aircrafts within a distance r_1 from the target, and n_2 is the number of aircrafts within a distance r_2 useful to fulfill a secondary task.

The last Utility function can be computed to find the best strategy in order to perform the surveillance of both target and the surrounding area. In this case the presence of one aircraft very close to the target and one (or more) away from it are both rewarded. The formula has the same analytical dependence of the former Utility (eq.11), but the rewarding is different.

IV. Surveillance Strategy

Once the tasks of the surveillance are "translated" in a proper objective function, a quantitative evaluation of the mission can be performed. So, the command center duty is to deploy the *optimal* set of control inputs to the aircrafts. From a mathematical point of view, "optimal" means to maximize the chosen objective function. More specifically, we can define the optimal *strategy* as the set of control inputs that maximizes the expected value of the function along the whole mission time.

The task of finding the optimal strategy can be often unfeasible, given the amount of variables involved in the problem. A strategy is indeed composed by two real-valued parameters (the components of the control input) for each aircraft and for each time interval. The calculation of the expected value of a strategy is particularly time consuming, due to the uncertainties in play.

In order to overcome the above difficulties, we defined a "heuristic" approach that can determine a "good" starting strategy. Small variations of it are applied until a (local) maximum is found. Of course, there is no guarantee that this approach actually can actually find the global maximum.

V. Results

In this section we report the main results obtained implementing the proposed model in a software R.

A. Target Tracking

In this subsection, a comparison between BN and KF is presented.

The target path is simulated for the whole mission time. Defining as t_p the "present" time, i.e. the time of the last revelation of the target position, it is possible to divide the mission time between the past ($t < t_p$) where the path of the target must be reconstructed from the measurements, and the future ($t > t_p$) where the KF and the BN has to predict the target future trajectory with a given tolerance.

The reconstruction of the target path is computed by means of both techniques. It has to be stressed that the BN approach is based on a Monte Carlo calculation, and so it can give slightly different results for each run. We verified that BN converge to the KF results as the number of simulations grows and both seem to correctly reproduce the target motion for a wide range of situations, provided that the covariance matrices of both motion and measurements noises are fully known. As we already discussed, the BN approach can be more general; however it has a couple of drawbacks. The first is the computational cost: calculations can be very heavy. The second is that small differences in the estimation of the state vector at the present time can make a big difference in the predictions of the future positions. Figure 2 shows a typical target tracking scenario.

B. Heuristic Strategy for Drones

As we discussed in Section IV, in order to shorten the calculation of finding the optimal strategy, we developed a heuristic approach aimed to rapidly find a "good" strategy. It consists in the following.

- Determine the most probable position of the target for each time. We call P the vector of these positions.

- We then determine the matrix S_{ij}, i ranging in [1,Nd] (Nd is the number of aircrafts) and j in [1,T], where T is the time left in the surveillance, in which the element s_{ij}, represents the set of commands that allows the aircraft i reach the target at the time j.

- To each aircraft is assigned one of the T set of commands determined above. A strategy is then defined and its expected utility evaluated.

- In this way NdxT possible strategies are defined; for all of them the expected utility is evaluated and the best one is the starting strategy.

Figure 3 shows the procedure in a case with N=3 aircrafts and T=6 times left in the surveillance, for the three utility functions defined in Section III. As we can see, a collaborative approach emerges naturally: aircrafts that are farther from the target try to intercept them at later stages, while closer aircrafts attempt to reach the target as soon as possible. The starting strategy may change accordingly to the utility function. Since the second and the third of them awards more than one aircraft to be close to the target, the respective strategies are more "eager" to reach the target.

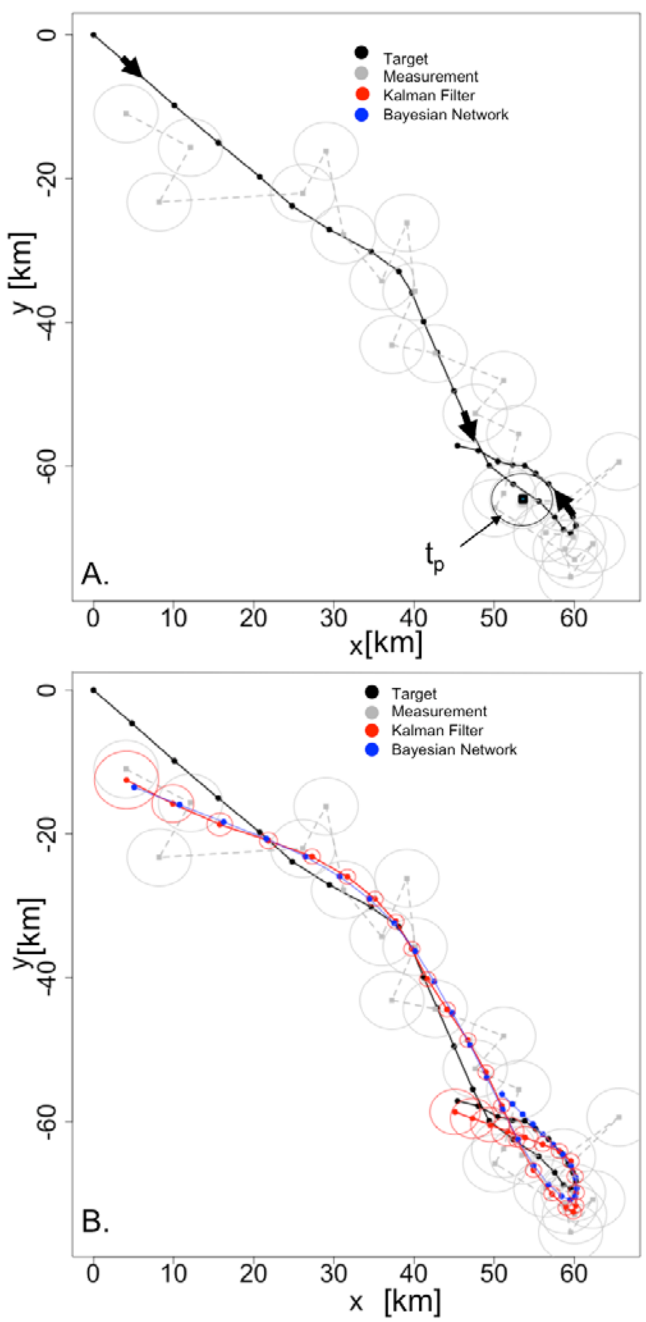

Fig. 2.(Top) Target motion (black) and measurements (gray) with uncertainty

are reported. Last measurement is revealed ad t=tp, whereas the motion of the target is simulated during the whole mission time (Bottom). Comparison between Kalman Filter (red) and Bayesian Network (blue) for the reconstruction of the target motion until t_p and prevision for the next five future times.

A strategy is defined by choosing for each aircraft one of the possible 6 control inputs that lead the aircraft as close as possible to the target at each of the remaining 6 time intervals. For each strategy the expected utility is computed and the best is selected. In Fig 4 we see the emerging of a collaborative approach: the aircrafts try to reach the target at different times in order to maximize the surveillance.

VI. Conclusion

This paper focused on a path planning strategy definition for system of multiple Unmanned Aircraft, i.e. a swarm, during surveillance missions. The developed method was based on the realization of a target tracking algorithm for target and drones motion estimation by means of a Kalman Filter and Bayesian network approach. After the tracking phase, a series of utility functions have been defined to support the decision making strategy. The main objective was to find a set of commands for the network of drones able to maximize the defined utility functions and the surveillance area during the entire mission. The results have shown the ability of the method to track the target in a wide range of situations and to define a good strategy for the deployment of the aircraft, thus creating a collaborative approach where aircraft closer to the target move in advance with respect to the ones more distant.

This paper represents a first step for a fully realistic description of the problem. For instance, we didn't take into account environmental effects that could significantly modify the motion abilities of the objects in play. We plan to expand this work by introducing external factors that modify the dynamics of the objects involved.

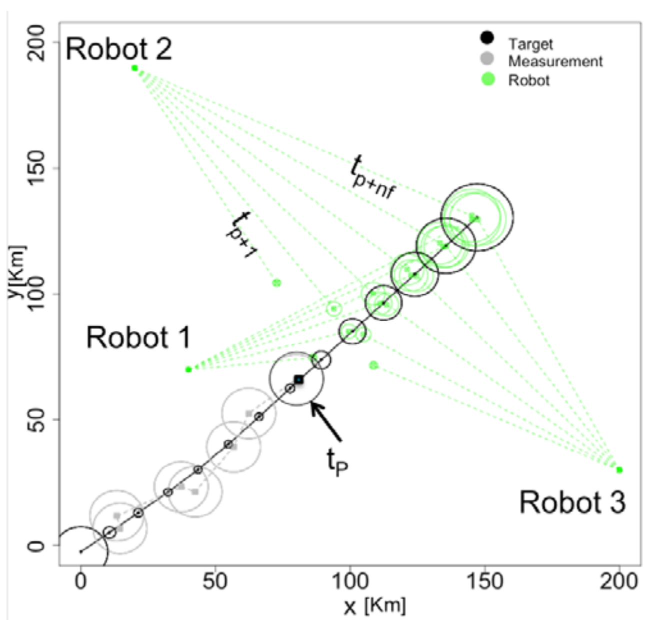

Fig. 3. Heuristic approach for finding a "good" aircraft deployment strategy. For each aircraft the relative control input needed to reach the target at a given time is computed. The green dashed lines in the figure represent the trajectories each aircraft follows as a consequence of the respective control input. The green circle is the one-sigma probability region of the aircraft position at the time it supposedly should reach the target.

A. *Commands for different Utility*

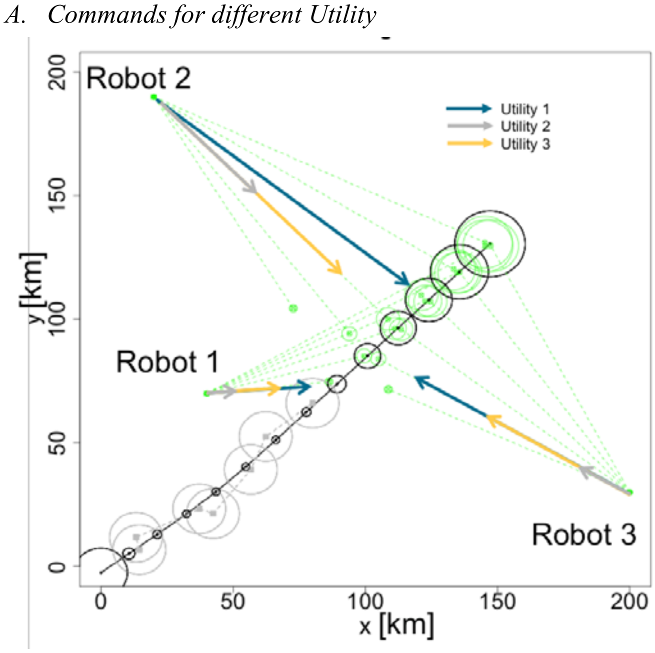

Fig. 4: Selected strategy for each of the three utility functions adopted (see Section(III)).

Acknowledgements

The idea of this work was conceived during the Decision Making School seminars funded by MBDA.

The authors would like to acknowledge Prof. Alfredo Germani from University of L'Aquila (Italy) for fruitful and enlightening discussions.

References

[1]. Luna and K. E. Bekris, "Efficient and complete centralized multi-robot path planning, "Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, San Francisco, CA, 2011, pp. 3268-3275. doi: 10.1109/IROS.2011.6095085.

[2]. Dimitra Panagou, Matthew Turpin and Vijay Kumar, Decentralized Goal Assignment and Trajectory Generation in Multi-Robot Networks: A Multiple Lyapunov Functions Approach, 2014 IEEE International Conference on Robotics & Automation (ICRA) Hong Kong Convention and Exhibition Center May 31 - June 7, 2014. Hong Kong, China.

[3]. A. Jimenez-Pacheco, D. Bouhired, Y. Gasser, J. C. Zufferey, D. Floreano and B. Rimoldi, "Implementation of a wireless mesh network of ultra light MAVs with dynamic routing, "Globecom Workshops (GC Wkshps), 2012 IEEE, Anaheim, CA, 2012, pp. 1591-1596. doi: 10.1109/GLOCOMW.2012.6477823.

[4]. M. Varga, M. Basiri, G. Heitz and D. Floreano, "Distributed formation control of fixed wing micro aerial vehicles for area coverage, "Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on, Hamburg, 2015, pp. 669-674. doi: 10.1109/IROS.2015.7353444.

[5]. G. De La Torre, T. Yucelen and E. N. Johnson, "Hybrid protocols for maintaining connectivity of multiagent systems," Decision and Control (CDC), 2013 IEEE 52nd Annual Conference on, Firenze, 2013, pp. 805-810. doi: 10.1109/CDC.2013.6759981.

[6]. Girish Chowdhary, Emilio Frazzoli, Jonathan P. How, Hugh Lui, Handbook of Unmanned Aerial Vehicles, "Nonlinear Flight Contol Techniques for Unmanned Aerial Vehicles", Springer, 2012.

[7]. Hwan-il Kang; Byunghee Lee; Kabil Kim, "Path Planning Algorithm Using the Particle Swarm Optimization and the Improved Dijkstra Algorithm," Computational Intelligence and Industrial Application, 2008. PACIIA '08. Pacific-Asia Workshop on , vol.2, no., pp.1002,1004, 19-20 Dec. 2008.

[8]. F. K. Purian and E. Sadeghian, "Mobile robots path planning using ant colony optimization and Fuzzy Logic algorithms in unknown dynamic environments," Control, Automation, Robotics and Embedded Systems (CARE), 2013 International Conference on, Jabalpur, 2013, pp. 1-6. doi: 10.1109/CARE.2013.6733718

[9]. Zhongya Wang, Min Li, Lianhang Dou, Yang Li, Qingying Zhao and Jie Li, "A novel multi-objective artificial bee colony algorithm for multi-robot path planning," Information and Automation, 2015 IEEE International Conference on, Lijiang, 2015, pp. 481-486. doi: 10.1109/ICInfA.2015.7279336.

[10]. Nandy, Sudarshan, Partha Pratim Sarkar, and Achintya Das. "Analysis of a Nature Inspired Firefly Algorithm based Back-propagation Neural Network Training." arXiv preprint arXiv:1206.5360 (2012).

[11]. Martyn Plummer (2015). rjags: Bayesian Graphical Models using MCMC. R package version 4-4. http://CRAN.R-project.org/package=rjags.

[12]. Giovanni Petris (2010). An R Package for Dynamic Linear Models. Journal of Statistical Software, 36(12), 1-16. URL http://www.jstatsoft.org/v36/i12/. Petris, Petrone, and Campagnoli. Dynamic Linear Models with R. Springer (2009).

[13]. David Luethi, Philipp Erb and Simon Otziger (2014). FKF: Fast Kalman Filter. R package version 0.1.3. http://CRAN.R-project.org/package=FKF