Asynchronous Distributed Camera Network Patrolling under Unreliable Communication

Nicoletta Bof, Ruggero Carli, Angelo Cenedese, and Luca Schenato

***Abstract*—In this paper we study the problem of real-time optimal distributed partitioning for perimeter patrolling in the context of multi-camera networks for surveillance, where each camera has limited mobility range and speed, and the communication is unreliable. The objective is to coordinate the cameras in order to minimize the time elapsed between two different visits of each point of the perimeter. We address this problem by casting it into a convex problem in which the perimeter is partitioned into non-overlapping segments, each patrolled by a camera that sweeps back and forth at the maximum speed. We then propose an asynchronous distributed algorithm that guarantees that these segments (1) cover the whole patrolling perimeter at any time and (2) asymptotically converge to the optimal centralized solution under reliable communication. We finally modify the proposed algorithm in order to attain the same convergence and covering properties even in the more challenging scenario where communication is lossy and there is no channel feedback, i.e. the transmitting camera is not aware whether a packet has been received or not by its neighbours.**

***Index Terms*—Patrolling, Camera Networks, Distributed Algorithms**

I. Introduction

VIDEO surveillance systems are nowadays increasingly used for security and prevention purposes in a variety of different situations. They can be used as a deterrent for intruders or for the early detection of anomalous events.

Important tasks required to these systems are target acquisition, tracking, activity recognition [1], [2], [3] and patrolling.

In this work we concentrate on the patrolling problem for networks of Pan-Tilt-Zoom (PTZ) cameras. This problem corresponds to the repetitive monitoring of a perimeter or of an area realized by a group of cameras, in order to be able to detect intruders or to locate unexpected events. We suppose to have a group of already deployed and fixed PTZ cameras that have to patrol a given one-dimensional environment.

The patrolling problem on a one-dimensional environment using a network of PTZ cameras is studied in [4], where it is reduced to a partitioning problem. This approach is effective in case the intruder is static. To deal with dynamic intruders, the partitioning has to be combined with a given schedule for the movements of the cameras, as shown in [5], [6]. For the patrolling of two-dimensional areas, randomized strategies have been proposed in [7], [8].

The patrolling problem for networks of PTZ cameras has similarities to that for mobile-agents, which is studied in many papers. With no intention of providing an exhaustive overview on the subject, some related literature is reported in the following. The problem of patrolling different disjoint areas using agents that can move from one area to the other is studied for example in [9] and [10]: the first solves it as a travelling salesman problem, while the latter uses a swarm intelligence approach. Different solutions are provided in [11], where reinforcement learning is adopted to deal with a similar problem and in [12], where the patrolling of a one-dimensional environment is addressed solving an optimal control problem. More interestingly with respect to the proposed work, the authors of [13] and [14] consider the patrolling of a one or two-dimensional environment and reduce this problem to a partitioning problem similar to that in [4]. The current trend in large-scale smart camera networks, including many of the articles just cited, is to devise distributed algorithms. These have many advantages in large networks, since they are scalable and require information only from neighbouring agents, allowing for privacy. Moreover, it may be difficult to collect all the information in a single unit if the communication is not reliable. A distributed approach may also be safer in presence of attackers, who would have to compromise each single camera and not just a central unit. Finally a distributed algorithm can adapt to dynamic scenarios in which cameras switch from patrolling mode to tracking mode and vice-versa, or in which some cameras may be malfunctioning.

All the previous algorithms assume reliable communications that might not be very realistic in wireless camera networks, which are becoming very popular thanks to their reduced installation and configuration cost and increased bandwidth performances. The major contribution of this work is to propose an asynchronous distributed algorithm for camera network patrolling that is guaranteed to converge to an optimal solution while ensuring certain properties even if communication is not reliable and there is no channel feedback, i.e. a transmitting camera is not aware whether a packet has been received or not by its adjacent neighbours. Devising coordinated cooperative algorithms under this scenario is particularly challenging since it implies that the cooperating cameras have no information on which action a neighbouring camera is doing, but has the advantage of requiring simple and fast communication protocols[1]. We incrementally tackle this scenario by first formulating the optimal centralized solution, then by providing a distributed algorithm that reaches the same solution asymptotically and, finally, by modifying it to handle the unreliable communication case.

All the authors are with the Department of Information Engineering, University of Padova, Via Gradenigo 6/a,35131 Padova, Italy. {bofnicol, carlirug, schenato}@dei.unipd.it, angelo.cenedese@unipd.it.

[1]Just as an example, even the apparently simple problem of computing an exact average over a network of smart agents with unreliable communication has been shown to be particularly difficult [15].

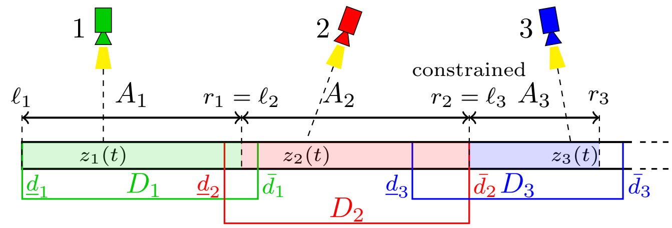

Fig. 1. Portion of the perimeter to be patrolled. The figure shows the physical coverages $\{D_i\}$ and the patrolling areas $\{A_i\}$ for the first three cameras of the surveillance system.

II. Perimeter patrolling: problem formulation

In this section we review the problem of patrolling a one-dimensional environment of finite length with a finite number of cameras, a situation typical of outdoor camera networks monitoring the boundary of an area of interest, such as urban neighbourhoods or large facility perimeters. Let $\mathcal{L} = [0, L]$, $L > 0$, denote the segment to be monitored and let N be the cardinality of the camera set, with the cameras labelled 1 through N. For the sake of simplicity, let us assume that (a) the cameras are 1-d.o.f., meaning that the field of view (f.o.v.) of each camera is allowed to change due to pan movements only, (b) the cameras have fixed coverage range, meaning that during pan movements the camera coverage range is not altered by the view perspective, (c) cameras have point f.o.v..

We define the *patrolling range* D_i as the total allowed area that the i-th camera can patrol due to the scenario topology, the agent configuration and its physical constraints. More formally

$$D_i = [\underline{d}_i, \overline{d}_i] \subset \mathcal{L}, \qquad \underline{d}_i < \overline{d}_i.$$

where $\underline{d}_i$, $\overline{d}_i$ are the left and the right extremes of the interval D_i, respectively. We assume that the patrolling ranges D_i, $i \in \{1, \ldots, N\}$, satisfy the following *interlacing* physical coverage constraints,

$$\underline{d}_i \le \underline{d}_{i+1} \le \overline{d}_i \le \overline{d}_{i+1}, \quad i = 1, \ldots, N-1. \tag{1}$$

Moreover we impose $\underline{d}_1 = 0$ and $\overline{d}_N = L$. These conditions guarantee $\cup_{i=1}^N D_i = \mathcal{L}$, i.e, the segment $\mathcal{L}$ can be fully patrolled. See also [16] for a further discussion on these issues.

The *max speed* $\bar{v}_i \in \mathbb{R}^+$ is the maximum speed of the i-th camera during pan movements, i.e., $|v_i(t)| \le \bar{v}_i$.

The *camera position* $z_i(t)$ is the position of the f.o.v. of the i-th camera as a function of the time variable t.

The *patrolling area* $A_i = [\ell_i, r_i]$, with $\ell_i \le r_i$, respectively the left and right extremes of A_i, denotes the area that is actively patrolled by the i-th camera and, differently from D_i, can be updated in time t, namely $A_i = A_i(t)$. The **physical constraints** $A_i \subseteq D_i$ must hold for all cameras at any time.

Note that the patrolling areas $\{A_i\}_{i=1}^N$ can be equivalently described by the pair of vectors $r = [r_1 \ \ldots \ r_n]$ and $\ell = [\ell_1 \ \ldots \ \ell_N] \in \mathbb{R}^N$, or by vector $\xi = [r^\top \ \ell^\top]^\top \in \mathbb{R}^{2N}$.

Figure 1 depicts a portion of the surveillance scenario.

To properly define the patrolling problem we need to introduce a suitable cost function and, accordingly, state an optimality criterion. After introducing the original problem in its exact form, we will modify it in order to have a different but much simpler one, whose solutions are suboptimal with respect to the former problem. For $x \in \mathcal{L}$ and $j \in \mathbb{N}$, let $\bar{\tau}_j = [\underline{t}_j(x), \bar{t}_j(x)]$ be the time interval during which the point x is visited for the $j-$th time (counting times starting from time $t = 0$) by at least a camera $i \in \{1, \ldots, N\}$, that is, for all $t \in \bar{\tau}_j$ there exists a camera i such that $z_i(t) = x$. If point x is visited at time t only by a passing camera i, i.e, $z_i(t) = x$ and $\dot{z}_i(t) \neq 0$, it holds $\underline{t}_j(x) = \bar{t}_j(x)$. Now, for each point $x \in \mathcal{L}$, we can introduce the following cost function

$$\gamma(x) := \begin{cases} \sup\limits_{j \in \mathbb{N}} \left(\underline{t}_{j+1}(x) - \bar{t}_j(x)\right) & \text{if } \forall j \quad \exists \underline{t}_{j+1}(x) < \infty \\ +\infty & \text{otherwise.} \end{cases}$$

The global cost function is given by the following time lag

$$T_{lag} = \sup_{x \in \mathcal{L}} \gamma(x)$$

and the problem we would like to solve is the minimization of T_{lag}, that is the minimization of the elapsed time between two consecutive visits of the same location of $\mathcal{L}$. To have $T_{lag} < \infty$ it is necessary that each point $x \in \mathcal{L}$ belongs to at least one patrolling area A_i, namely, that the **covering constraint** $\bigcup_{i \in \{1,\ldots,N\}} A_i = \mathcal{L}$ is satisfied. Observe that, if the following **interlacing constraints**

$$\ell_i < \ell_{i+1} \le r_i < r_{i+1}, \ \forall i = 1, \ldots, N \tag{2}$$

are satisfied, then also the covering constraint is satisfied[2].

The minimization of T_{lag} in case there are no physical constraints for the cameras is a tricky problem. As a matter of fact, one could reasonably think of using a partitioning approach to solve the problem as done in [17], where the following conjecture is given:

Conjecture II.1 *Assume $D_i = \mathcal{L}$ for all i. Then the optimal minimum value for T_{lag} is attained by partitioning $\mathcal{L}$ into non-overlapping intervals of lengths proportional to the cameras speeds, specifically*

$$\ell_1 = 0, \qquad r_N = L, \qquad r_i = \ell_{i+1} = \ell_i + \frac{\bar{v}_i}{\sum_{i=1}^N \bar{v}_i} L,$$

and letting each camera i sweeping back and forth A_i at its maximum pan speed $\bar{v}_i$. This strategy obtains

$$T_{lag} = \frac{2L}{\sum_{i=1}^N \bar{v}_i}. \tag{3}$$

The above conjecture has been shown to be true in the following two scenarios [18]:

(i) when $N = 1, 2, 3$;
(ii) for any $N > 3$ in case $\bar{v}_1, \ldots \bar{v}_N$ are all equal to each other (i.e, there exist $\bar{v}$ such that $\bar{v}_1 = \cdots = \bar{v}_N = \bar{v}$).

Remarkably, in case the maximum pan speeds are not all equal to each other, the authors in [18] have showed the existence of some particular N-uplas $(\bar{v}_1, \ldots, \bar{v}_N)$ $(N > 3)$, for which it is possible to design cameras' trajectories attaining a value of T_{lag} smaller than that in (3), thus invalidating Conjecture II.1.

[2]To be precise, to satisfy the covering constraint also the **boundary constraints** $\ell_1 = 0$ and $r_N = L$ needs to be satisfied. The standing assumption in this paper is that they are always satisfied.

Despite the presence of these counterexamples, in [18] it is however argued that the solution illustrated in [17] attains a value of T_{lag} that is very close to the optimal one, that is, it can be regarded as a significant sub-optimal solution.

Therefore, since the very simple *"partitioning and sweeping back and forth at maximum speed"* strategy described in Conjecture II.1 is likely to be an almost optimal solution of the patrolling problem, we change the set of possible cameras' trajectories: instead of minimizing the patrolling time lag among all possible trajectories, we restrict them. In particular we force a partition of the environment and we have each camera sweep its own part of the perimeter at maximum speed. Our problem is then to find the partition that minimizes T_{lag} under this restriction of the trajectories. Therefore, the problem we want to tackle is a partitioning one. Apart from its semplicity and suboptimality, the choice of partitioning acquires even more significance in our scenario, since we include also the presence of physical constraints, which might impose severe limitations to the areas to be patrolled by the cameras (and therefore lead to more complex trajectories).

To formally state the optimization problem we are interested in, we preliminarily observe that, by sweeping back and forth at speed $\bar{v}_i$ a given interval $A_i = [\ell_i, r_i]$, it holds that the time lag for camera i, $i = 1, \dots, N$, is $T_{lag}(A_i) = \frac{2|A_i|}{\bar{v}_i}$, where $|A_i| := r_i - \ell_i$. Then, the problem we aim at solving can be cast as

$$\mathcal{P}_1: \quad T^*_{\mathcal{P}_1} = \min_{A_1,\dots,A_N} \max_i \{T_{lag}(A_i)\}$$
$$s.t. \begin{cases} A_i \subseteq D_i, & i = 1, \dots, N \\ \cup_{i=1}^N A_i = \mathcal{L} & \end{cases}$$

where the objective is the minimization of the largest patrolling time lag among all areas A_i, and the constraints represent the physical limitations of the cameras and the requirement that all points in $\mathcal{L}$ are eventually visited, respectively. The previous problem can be re-cast as a linear program (LP) as follows (the proof can be found in [4]):

Proposition II.2 *[4] The optimization problem $\mathcal{P}_1$ is equivalent to the following LP problem:*

$$\mathcal{P}'_1: \quad T^*_{\mathcal{P}_1} = \min_{\xi,\tau} 2\tau$$
$$s.t. \begin{cases} \frac{r_i - \ell_i}{\bar{v}_i} \le \tau & i = 1, \dots, N \\ \underline{d}_i \le \ell_i \le \overline{d}_i, \quad \underline{d}_i \le r_i \le \overline{d}_i & i = 1, \dots, N \\ r_i \ge \ell_{i+1} & i = 1, \dots, N \\ \underline{d}_1 = \ell_1 = 0, \quad \overline{d}_N = r_N = L & \end{cases}$$

Analysing the previous problem, we can introduce the following cost functional

$$J_\infty(\xi) = \max_i \frac{2(r_i - \ell_i)}{\bar{v}_i},$$

for which it holds that the minimum value J^*_∞ achievable for $J_\infty(\xi)$, with ξ respecting the physical and interlacing constraints, is equal to the optimal solution $T^*_{\mathcal{P}_1}$ of problem $\mathcal{P}'_1$. We denote by $\mathcal{E}^*_{\mathcal{P}_1}$ the set of minimizers of $J_\infty(\xi)$ or, equivalently, of $\mathcal{P}'_1$.

The previous proposition provides a centralized solution to the patrolling problem, but cannot be easily computed in a distributed fashion. Although distributed algorithms exist for the solution of LP problems [19], these involve the solution of the entire problem at each node, which is a futile computational effort. Moreover, the previous optimization problem might have multiple minimizers. Again, we can formulate a new optimization problem $\mathcal{P}_2$, whose minimizer is unique and is also a minimizer for the original problem $\mathcal{P}_1$. Introducing the following cost functional

$$J_2(\xi) = \sum_{i=1}^N \frac{1}{\bar{v}_i}(r_i - \ell_i)^2, \tag{4}$$

we can state the following proposition (its proof is in [4])

Proposition II.3 *[4] Consider the optimization problem*

$$\mathcal{P}_2: \quad J^*_2 = \min_{\xi \in \mathbb{R}^{2N}} J_2(\xi)$$
$$s.t. \begin{cases} \underline{d}_i \le \ell_i \le \overline{d}_i, \quad \underline{d}_i \le r_i \le \overline{d}_i & i = 1, \dots, N \\ r_i \ge \ell_{i+1} & i = 1, \dots, N-1 \\ \underline{d}_1 = \ell_1 = 0, \quad \overline{d}_N = r_N = L & \end{cases}$$

The corresponding set of minimizers $\mathcal{E}^*_{\mathcal{P}_2}$ is a singleton and $\mathcal{E}^*_{\mathcal{P}_2} \subseteq \mathcal{E}^*_{\mathcal{P}_1}$.

The benefits of the optimization problem $\mathcal{P}_2$ as compared to the optimization problem $\mathcal{P}_1$ are mainly two, namely: (i) using specific communication strategies $\mathcal{P}_2$ can be solved with distributed, scalable and parallelizable algorithms; (ii) the uniqueness of the minimizer in $\mathcal{P}_2$ guarantees the practical convergence of iterative numerical algorithms.

Remark II.4 Note that, intuitively speaking, the solution of problem $\mathcal{P}_2$ shares the patrolling burden as evenly as possible among all the cameras. The unique partition that solves $\mathcal{P}_2$ is such that each camera has a time lag that is as similar as possible to the time lag of its neighbours[3].

Remark II.5 As pointed out in [5], having the cameras sweep the assigned portions of the perimeter at the maximum speed is efficient for static intruders, while for smart dynamic intruders a more sophisticated law is needed. However, this control law, which only involves the synchronization of the movements of the cameras, can be applied to any partitioning of the environment to be patrolled. Therefore, for smart intruders we can apply the equal-waiting trajectory algorithm suggested in [5] on the partitioning of $\mathcal{L}$ given by the optimal solution $\mathcal{E}_{\mathcal{P}_2}$ of problem $\mathcal{P}_2$. In this work we focus on the optimal partitioning of the environment, and, by applying the equal-waiting trajectory to our optimal partitioning, we can combine good performance both for static and dynamic intruders.

Remark II.6 Problems $\mathcal{P'}_1$ and $\mathcal{P}_2$ can be rapidly solved using a centralized algorithm. However, a distributed approach may be more advisable due to the advantages already highlighted in the introduction, and in particular those regarding the

[3] In some way, it is similar to what happens with the problem of finding x such that $Ax = b$, with $A \in \mathbb{R}^{n \times n}$ singular and $b \in \mathbb{R}_n$ a given vector. The problem has many solutions, but the one obtained by using the pseudo inverse of A is the one that minimizes the norm of vector x.

capability to adapt to dynamic changes like intruders tracking or the presence of faulty cameras [5]. Note that these events are usually only local, and in a centralized approach, each time a new intruder appears or each time a camera fails the algorithm has to be reset for the whole network.

III. A COORDINATED BROADCAST PARTITIONING ALGORITHM (*CB* ALGORITHM)

Our goal is to design an iterative partitioning algorithm that allows the cameras to update their patrolling areas using only information coming from neighbouring cameras (camera i, $i \in \{2, \dots, N-1\}$, exchanges information with camera $i-1$ and camera $i+1$; if $i = 1$ (resp. $i = N$) the only neighbour of camera 1 (resp. N) is camera 2 (resp. $N-1$)). The update of the patrolling area has to be such that (i) the physical constraints and the covering constraint are satisfied at each iteration, and (ii) the set of patrolling areas converges to the optimal partition.

We assume that the information flow is regulated by some asynchronous communication protocol, namely, that there is not a common reference time that keeps all the updating and transmitting actions synchronized among all the cameras. It is well known that synchronous communications are less robust to delays, packet losses and interference phenomena. In particular, the communication protocol we adopt is a *coordinated broadcast* (ideally, the transmitting camera communicates with both its neighbours at the same time, and these reply back immediately after).

The strategy we propose is next described and reported as Algorithm 1. Suppose the patrolling areas are initialized in such a way that the physical and interlacing constraints are satisfied and let the iterations of the algorithm be indexed by the discrete time variable $t \in \mathbb{N}$. Then, at iteration t, one camera[4], say i, is activated, that is it performs one iteration of the algorithm. Camera i transmits the values of $\ell_i(t)$ and $r_i(t)$ to its neighbouring cameras $i-1$, $i+1$. Based on the information received, cameras $i-1$ and $i+1$ update the extremes of their patrolling areas that are "closer" to camera i, namely, r_{i-1} and ℓ_{i+1}, respectively. For simplicity, we consider only the update performed by camera $i-1$.

Let $m_{i-1}(t)$ and $m_i(t)$ be the middle points of $A_{i-1}(t)$ and $A_i(t)$, respectively, i.e,

$$m_{i-1}(t) = \frac{\ell_{i-1}(t) + r_{i-1}(t)}{2}, \quad m_i(t) = \frac{\ell_i(t) + r_i(t)}{2}. \quad (5)$$

Camera $i-1$ computes the point c^*_ℓ which splits the segment $[m_{i-1}(t), m_i(t)]$ into two parts that require the same time to be swept by the respective cameras; the formal expression for c^*_ℓ is given in line 3 of Algorithm 1.

Camera $i-1$ sets $r_{i-1}(t+1) = c^*_\ell$, provided that this update does not violate the physical constraints, i.e, it must holds $c^*_\ell \in [\underline{d}_i, \overline{d}_{i-1}]$; otherwise $r_{i-1}(t+1)$ is set equal to the closest point to c^*_ℓ that satisfies the physical constraint (see lines 4 through 10). Finally camera $i-1$ sends the value $r_{i-1}(t+1)$ to camera i which updates its left extreme accordingly, that is,

[4]In the description of the algorithms, we suppose that the selected camera at time t is $i = 2, \dots, N-1$; if $i = 1$ ($i = N$) an *ad hoc* adjustment has to be done, i.e. only the update of camera $i+1$ (resp. $i-1$) has to be done.

Algorithm 1 *CB* algorithm (time t, camera i activated)

1: Broadcast forward communication: camera i transmits $r_i(t)$ and $\ell_i(t)$ to cameras $i+1$ and $i-1$.
2: $\{$% Update of the right extreme of camera $i-1\}$
3: $c^*_\ell = \frac{\bar{v}_i(\ell_{i-1}(t)+r_{i-1}(t))+\bar{v}_{i-1}(r_i(t)+\ell_i(t))}{2(\bar{v}_i+\bar{v}_{i-1})}$;
4: **if** $c^*_\ell < \underline{d}_i$ **then**
5: $\quad r_{i-1}(t+1) = \underline{d}_i$;
6: **else if** $c^*_\ell > \overline{d}_{i-1}$ **then**
7: $\quad r_{i-1}(t+1) = \overline{d}_{i-1}$;
8: **else**
9: $\quad r_{i-1}(t+1) = c^*_\ell$;
10: **end if**
11: $\{$% Update of the left extreme of camera $i+1\}$
12: $c^*_r = \frac{\bar{v}_{i+1}(\ell_i(t)+r_i(t))+\bar{v}_i(\ell_{i+1}(t)+r_{i+1}(t))}{\bar{v}_i+\bar{v}_{i+1}}$;
13: **if** $c_r* > \overline{d}_i$ **then**
14: $\quad \ell_{i+1}(t+1) = \overline{d}_i$;
15: **else if** $c^*_r < \underline{d}_{i+1}$ **then**
16: $\quad \ell_{i+1}(t+1) = \underline{d}_{i+1}$;
17: **else**
18: $\quad \ell_{i+1}(t+1) = c^*_r$;
19: **end if**
20: $\{$% Update of the extremes of camera $i\}$
21: Peer to peer backward communication: camera i receives from its neighbours $\ell_{i+1}(t+1)$ and $r_{i-1}(t+1)$.
22: $\ell_i(t+1) = r_{i-1}(t+1)$;
23: $r_i(t+1) = \ell_{i+1}(t+1)$;

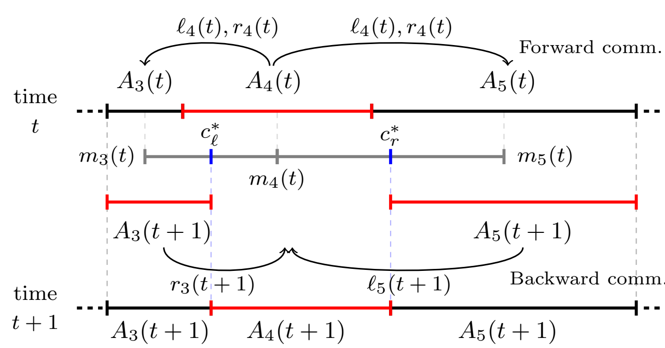

Fig. 2. Execution of one step of the algorithm in a simplified set-up with $D_i = \mathcal{L}$ and equal $\bar{v}_i$ for all i. The camera activated at time t is camera 4.

$\ell_i(t+1) = r_{i-1}(t+1)$ (see line 22). Camera $i+1$ carries out an analogous update: in this case $\ell_{i+1}(t+1)$ and $r_i(t+1)$ are the extremes involved (see lines 12 through 19 and line 23).

Figure 2 shows one step of the execution of the algorithm. Observe that each iteration of the *CB* algorithm involves two communication rounds; the first one from camera i to cameras $i-1$ and $i+1$, referred to as the *forward communication*, and the second one from cameras $i-1$ and $i+1$ to camera i, referred to as *backward communication*.

To characterize the convergence of the *CB* algorithm we need some condition on how often each camera is activated to perform the broadcast forward communication. Therefore we introduce the following property

Assumption III.1 (Persistent camera activation) *For all $t \in \mathbb{N}$, there exists a unique positive integer number τ such that within the interval $[t, t+\tau)$ each camera performs a broadcast forward communication at least once.*

We have the following result characterizing the convergence properties of the *CB* algorithm.

Theorem III.2 *Let $\xi(0)$ describe the initial patrolling areas, satisfying the physical and interlacing constraints. Assume Assumption III.1 holds true. Then the trajectory $t \to \{\xi(t)\}$ generated by the* CB *algorithm satisfies that*

(i) *the physical, interlacing and covering constraints are verified for all $t \in \mathbb{N}$;*

(ii) *the cost functional J_2 is not increasing and satisfies*

$$J_2(\xi(t+1)) < J_2(\xi(t)), \qquad \text{if } \xi(t+1) \neq \xi(t).$$

(iii) *the cost functional J_∞ is not increasing and satisfies*

$$J_\infty(\xi(t+\bar{\tau})) < J_\infty(\xi(t)), \qquad \text{if } \xi(t) \notin \mathcal{E}^*_{\mathcal{P}_1},$$

where $\bar{\tau} = (N-1)(\tau+1)$;

(iv) *the cost functionals J_2 and J_∞ converges, respectively, to J_2^* and J_∞^*.*

The proof is provided in Appendix A. The following result follows directly from item (iv) of the previous Theorem.

Corollary III.3 *Under the same hypotheses of Theorem III.2, the trajectory $t \to \{\xi(t)\}$ generated by the* CB *algorithm converges to the optimal solution of Problem $\mathcal{P}_2$, i.e, $\xi(t) \to \xi_2^*$, and, in turn, to an optimal solution of $\mathcal{P}_1$.*

IV. *r-CB*: ROBUSTIFICATION OF THE *CB* ALGORITHM TO PACKET LOSSES

In the previous section we have introduced the *CB* algorithm assuming that the communication channels are reliable and, in particular, that no packet losses occur. In this section we relax this assumption and we allow for transmission failures in the communication between neighbouring cameras. In presence of unreliable communications, the *CB* algorithm presents a major shortcoming, as explained in the following. Observe that, during each iteration of the *CB* algorithm, there are two possible "sources" of packet loss: (i) the packet broadcast by camera i during the forward communication is not received by camera $i-1$ (or analogously by camera $i+1$); in this case, the respective extremes remain unchanged and nothing happens; (ii) the packet sent by camera $i-1$ (or analogously by $i+1$) to camera i during the backward communication is not received, and, in turn, camera i does not update the respective extreme; it might result that $r_{i-1}(t+1) \neq \ell_i(t+1)$ and the interlacing and covering constraints might be violated (see Figure 3).

Observe that the latter failure is the most critical one; indeed it might cause the presence of parts of the perimeter that are not assigned to any of the cameras. To deal with such presence of uncovered areas, we provide a modification of the *CB* algorithm. Specifically, consider iteration t and assume that the interlacing constraints (2) among all the cameras are satisfied. Moreover assume that camera i is the camera performing the

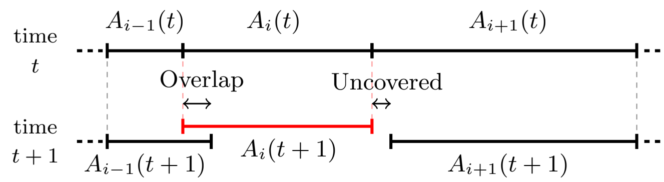

Fig. 3. Consequences of the failure of the backward communication: generation of an overlap between the patrolling areas and of an uncovered part of the environment. The situation at time t is corresponding to that presented in Figure 2.

Algorithm 2 *r-CB* algorithm (time t, camera i activated)

Broadcast forward communication: camera i transmits $r_i(t)$ and $\ell_i(t)$ to cameras $i+1$ and $i-1$.
{% Update if camera $i-1$ receives information}
$c_\ell^* = \frac{\bar{v}_i(\ell_{i-1}(t)+r_{i-1}(t))+\bar{v}_{i-1}(r_i(t)+\ell_i(t))}{2(\bar{v}_i+\bar{v}_{i-1})}$;
if $c_\ell^* < \ell_i(t)$ **then**
  $r_{i-1}(t+1) = \ell_i(t)$;
else
  $r_{i-1}(t+1) = \min\{c_\ell^*, \bar{d}_{i-1}\}$;
end if
{% Update if camera $i+1$ receives information}
$c_r^* = \frac{\bar{v}_{i+1}(\ell_i(t)+r_i(t))+\bar{v}_i(\ell_{i+1}(t)+r_{i+1}(t))}{\bar{v}_i+\bar{v}_{i+1}}$;
if $c_r* > r_i(t)$ **then**
  $\ell_{i+1}(t+1) = r_i(t)$;
else
  $\ell_{i+1}(t+1) = \max\{c_r^*, r_i(t)\}$;
end if
The algorithm performs steps $20 \div 23$ of Algorithm 1, provided the backward communications are successful.

forward communication round. If camera $i-1$ receives the information related to $\ell_i(t)$ and $r_i(t)$, then it computes c_ℓ^* as done for the *CB* algorithm, and it updates r_{i-1} as follows

$$r_{i-1}(t+1) = \begin{cases} \ell_i(t) & \text{if } c_\ell^* \leq \ell_i(t) \\ \min\{c_\ell^*, \bar{d}_{i-1}\} & \text{if } c_\ell^* > \ell_i(t) \end{cases} \tag{6}$$

Then camera $i-1$ sends the value $r_{i-1}(t+1)$ to camera i; if the packet is received, then camera i sets $\ell_i(t+1) = r_{i-1}(t+1)$, otherwise ℓ_i remains unchanged, i.e., $\ell_i(t+1) = \ell_i(t)$. Observe that, according to the update proposed in (6), it holds that $\ell_i(t+1) \leq r_{i-1}(t+1)$, and, hence, the interlacing constraint between cameras $i-1$ and i is still satisfied. This new algorithm, that is robust to packet losses, is denoted hereafter as *r-CB* (see the algorithmic description in Algorithm 2).

To characterize the convergence properties of *r-CB*, we need an Assumption on the frequencies of transmission failures.

Assumption IV.1 (Persistent communication) *Given any camera i, there exists a positive integer number h such that the number of consecutive failures in the forward communication from node i to node $i-1$ (or $i+1$) is smaller than h.*

We have the following result.

Theorem IV.2 *Let $\xi(0)$ describe the initial patrolling areas, satisfying the physical and interlacing constraints, and let Assumptions III.1 and IV.1 hold true. Then, the evolution $t \to \xi(t)$ generated by the* r-CB *algorithm satisfies that*

(i) *the physical, interlacing and covering constraints are verified for all $t \in \mathbb{N}$;*

(ii) *the cost functional $J_\infty(t)$ is not increasing and satisfies*

$$J_\infty(\xi(t+\tau_{\max})) < J_\infty(\xi(t)) \qquad \text{if } \xi(t) \notin \mathcal{E}^*_\infty.$$

where $\tau_{\max} := 2h\tau(N-1)+1$.

(iii) *$J_\infty(\xi(t))$ converges to J^*_∞.*

The proof is reported in Appendix B. The following result follows directly from the previous Theorem.

Corollary IV.3 *Under the same assumptions of Theorem IV.2, the trajectory $t \to \xi(t)$ converges to the set of optimal solution of problem $\mathcal{P}_1$, i.e., $\xi(t) \to \mathcal{E}^*_\infty$.*

Remark IV.4 The algorithm presented in this work is similar to the one presented in [4], [6]. However, the mathematical machinery used here is substantially different from the one employed in [4], [6], which considers only the lossless scenario and which strongly relies on the monotonicity of $J_2(\xi)$ and on its minimum being unique. In fact, when packet loss is considered neither $J_2(\xi)$ nor $J_\infty(\xi)$ satisfy the hypotheses of the theorems in [6]. Moreover, Theorem IV.2 is rather general and might be applicable to other relevant applications such as 2D/3D partitioning in cooperative robotics with asynchronous and lossy communication.

We carried out some simulations to further show the effectiveness of the algorithm. The setting is the following: the number N of cameras takes different values, the length of the environment is $L = 10N$ and the maximum speed is $\bar{v}_i = 2$ for all cameras. The patrolling range of cameras $i = 2, \ldots, N-1$ is $[10(i-1)-2, 10i+2]$, for camera 1 is $[0, 12]$ and for camera N is $[10(N-1)-2, 10N]$. Concerning the communication reliability, a communication works with a probability of 70% and the value for threshold h is 10 (in the implementation we assure that Assumption IV.1 is satisfied); also, every N iterations all the cameras are activated, implying a value for parameter τ in Assumption III.1 equal to $2N-1$. The initialization for the algorithm is $l_i(0) = \underline{d}_i$, $r_i(0) = \overline{d}_i(0)$ for all the cameras. Figure 4 shows the normalized cost functions for a realization of the *r-CB* algorithm. Given a cost function $J(t)$ with optimal value J^*, its normalized form $\tilde{J}$ is given by

$$\tilde{J}(t) = \frac{J(t) - J^*}{J(0) - J^*}. \tag{7}$$

The figure confirms that, as demonstrated in Theorem IV.2, $\tilde{J}_\infty$ does not increase as the number of iterations increases and converges to the optimal value, while for $\tilde{J}_2$ the non-increasing property does not hold. This clearly shows that $J_2(t)$ cannot be used as a Lyapunov function. Nevertheless, $J_2(t)$ still converges to its optimal value. Note that the speed of convergence of the algorithm is similar for all the sizes of

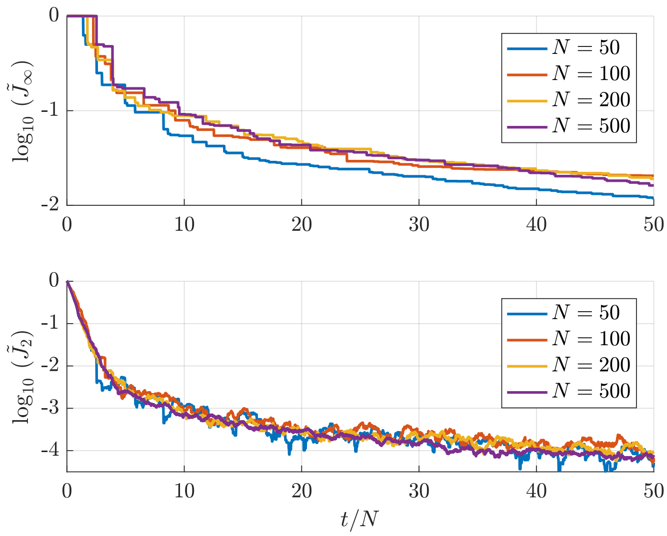

Fig. 4. Logarithm of the normalized cost functions $\tilde{J}_\infty$ and $\tilde{J}_2$ (see Equation (7)) for one realization of the *r-CB* algorithm and different values of N. The time scale is divided by N, which implies that at each integer value of t/N all the cameras have been activated once.

the network considered, which implies that the algorithm is very scalable. Together with scalability, we also underline the the algorithm has a really moderate computational burden.

V. Conclusion

In this paper we studied the patrolling problem, showing that, at least for static intruders, the partitioning of the environment is a very effective and simple strategy to solve the problem. We developed a distributed algorithm to allow each camera to compute its optimal patrolling area and we show that under reliable communication it converges to the optimal solution. In case of unreliable communication we propose another algorithm which, always respecting the (essential) coverage constraint, solves the partitioning problem. The latter is a particularly interesting analysis, which is not present in previous papers on the same subject.

Appendix A

Proof of Theorem III.2: **Point (i)** can be easily verified by analysing the steps of the algorithm. For **point (ii) and (iii)** we prove only the part concerning J_2 (the part of point (ii) related to J_∞ follows from Theorem IV.2). We start by observing that after τ iterations of the *BC* algorithm, we have that $r_i(t) = \ell_{i+1}(t)$ for all $i = 1, \ldots, N$. For $t \geq \tau$, we can introduce the auxiliary variables $x_i(t) = r_i(t) = \ell_{i+1}(t)$, $i = 1, \ldots, N-1$, and let $x(t)$ be the vector collecting all $x_i(t)$. The vector $x(t)$ represents the state of our system. The goal is to apply Theorem 4.3 of [20]. To do so, we will verify that all the hypotheses of this Theorem are satisfied in our context.

First of all, observe that, according to the physical constraints, we have that $\underline{d}_{i+1} \leq x_i(t) \leq \overline{d}_i$ and so $x(t)$ can take values only in $W = \prod_{i=1}^{N-1} \left[\underline{d}_{i+1}, \overline{d}_i\right]$. Since W is given by the Cartesian product of $N-1$ closed intervals, it follows that W is compact. Next, for $i \in \{1, \ldots, N\}$, let $T_i : W \to W$ be the map describing the updating iteration of CB algorithm in case camera i is the camera performing the forward communication round. Observe that, for $i \in \{1, \dots, N\}$, the map T_i is continuous with respect to the standard Euclidean metric. Now, for $x = [x_1, \dots, x_{N-1}] \in W$, let us introduce the function $U : W \to \mathbb{R}$ such that

$$U(x(t)) = \frac{1}{2}\sum_{i=1}^{N} \frac{L_i(t)^2}{\bar{v}_i}, \tag{8}$$

where $L_i(t) = r_i(t) - \ell_i(t) = x_i(t) - x_{i-1}(t)$. We need to show that U is a Lyapunov function for the update of the algorithm, i.e, that $U(x(t+1)) < U(x(t))$ whenever $x(t+1) \neq x(t)$. We start stating the following

FACT. *Let L, α, β be three positive real numbers. Then the minimizer of the function $g(x) = \frac{x^2}{\alpha} + \frac{(L-x)^2}{\beta}$ within the interval $[0, L]$ is given by $x = \frac{\alpha L}{\alpha+\beta}$.*

Suppose that at time t the $i-$th camera is activated, $i \neq 1$ and $i \neq N$, and consider the following sum of terms:

$$\Gamma = \frac{1}{2}\frac{L_{i-1}(t)^2}{v_{i-1}} + \frac{1}{2}\frac{L_i(t)^2}{v_i} + \frac{1}{2}\frac{L_{i+1}(t)^2}{v_{i+1}} =$$
$$= \frac{1}{v_{i-1}}\left(\frac{L_{i-1}(t)}{2}\right)^2 + \underbrace{\frac{1}{v_{i-1}}\left(\frac{L_{i-1}(t)}{2}\right)^2 + \frac{1}{v_i}\left(\frac{L_i(t)}{2}\right)^2}_{\gamma_1} +$$
$$+ \underbrace{\frac{1}{v_i}\left(\frac{L_i(t)}{2}\right)^2 + \frac{1}{v_{i+1}}\left(\frac{L_{i+1}(t)}{2}\right)^2}_{\gamma_2} + \frac{1}{v_{i+1}}\left(\frac{L_{i+1}(t)}{2}\right)^2.$$

Recalling the quantities in Formulas (5) and the value of $c_\ell^*(t)$, if we consider $\tilde{L}_{i-1}(t) = \frac{L_{i-1}(t)}{2} + \frac{L_i(t)}{2} = m_i(t) - m_{i-1}(t)$, the point $c_\ell^*(t) - m_{i-1}(t)$ is the minimizer of function $g(z)$ of parameters $L = \tilde{L}_{i-1}(t)$, $\alpha = v_{i-1}$ and $\beta = v_i$, as can be verified by calculation.

Introduce now $L'_{i-1}(t) = r_{i-1}(t+1) - m_{i-1}(t)$ and $L'_i(t) = m_i(t) - r_{i-1}(t+1) = \tilde{L}_{i-1}(t) - L'_{i-1}(t)$.

Since we have that $r_{i-1}(t) \le r_{i-1}(t+1) \le c_\ell^*$ or $r_{i-1}(t) \ge r_{i-1}(t+1) \ge c_\ell^*$, the update implies that

$$\gamma_1 = g\left(\frac{L_{i-1}(t)}{2}\right) \ge g(L'_{i-1}(t)).$$

A similar reasoning holds considering the update of camera $i+1$. According to the latter, defining $L''_{i+1}(t) = m_{i+1}(t) - \ell_{i+1}(t+1)$ and $L''_i(t) = \ell_{i+1}(t+1) - m_i(t)$, it holds

$$\gamma_2 \ge \frac{1}{v_i}(L''_i(t))^2 + \frac{1}{v_{i+1}}(L''_{i+1}(t))^2.$$

As a consequence, we have

$$\Gamma \ge \underbrace{\frac{1}{v_{i-1}}\left(\frac{L_{i-1}(t)}{2}\right)^2 + \frac{1}{v_{i-1}}(L'_{i-1}(t))^2}_{\delta_1} +$$
$$+ \underbrace{\frac{1}{v_i}(L'_i(t))^2 + \frac{1}{v_i}(L''_i(t))^2}_{\delta_2} + \underbrace{\frac{1}{v_{i+1}}(L''_{i+1}(t))^2 + \frac{1}{v_{i+1}}\left(\frac{L_{i+1}(t)}{2}\right)^2}_{\delta_1}.$$

Now define $L_{i-1}(t+1) = \frac{L_{i-1}(t)}{2} + L'_{i-1}(t)$, $L_{i+1}(t+1) = \frac{L_{i+1}(t)}{2} + L''_{i+1}(t)$ and $L_i(t+1) = L'_i(t) + L''_i(t)$. Analysing δ_1, we have that

$$\delta_1 = \frac{1}{v_{i-1}} g\left(\frac{L_{i-1}(t)}{2}\right) \ge \frac{1}{v_{i-1}} g\left(\frac{L_{i-1}(t+1)}{2}\right)$$

where in this case the parameters of function g are $L = L_{i-1}(t+1)$, $\alpha = \beta = 1$. A similar reasoning holds for δ_2 and δ_3. All the previous considerations lead to the following

$$U(x(t)) \ge \sum_{\substack{j=1 \\ j \neq i, i-1, i+1}}^{N} \frac{L_j(t)^2}{2} + 2\frac{1}{v_{i-1}}\left(\frac{L_{i-1}(t+1)}{2}\right)^2 +$$
$$+ 2\frac{1}{v_i}\left(\frac{L_i(t+1)}{2}\right)^2 + 2\frac{1}{v_{i+1}}\left(\frac{L_{i+1}(t+1)}{2}\right)^2 = U(x(t+1)).$$

When the camera that is activated at time t is the 1-st or the $N-$th a similar reasoning shows that $U(x(t)) \ge U(x(t+1))$. Therefore we have $U(x(t)) \ge U(x(t+1))$, and the inequality is strict as long as at least one of the following holds, $l_{i+1}(t+1) \neq l_{i+1}(t)$, $r_{i-1}(t+1) \neq r_{i-1}(t)$.

We are now in the position of applying Theorem 4.3 of [20] and to conclude that $x(t)$ converges to the set $F_1 \cap \cdots \cap F_N$, where $F_i = \{x \in W \mid T_i(x) = x\}$ is the set of fixed points of T_i. Clearly $F_1 \cap \cdots \cap F_N$ is a singleton that coincides with the optimum of problem $\mathcal{P}_3$. Since $J_2(t)$ converges to J_2^*, necessarily also $J_\infty(t)$ converges to J_∞^*. ■

APPENDIX B

The following Theorem is a refinement of Theorem 4.3 in [20], valid for a specific class of dynamical switching systems.

Theorem B.1 *Let $W \subset \mathbb{R}^n$ be a compact set. Let m be a positive integer and let $\{T_i : W \to W,\ i = 1, \dots, m\}$ be a set of m continuous functions. Let $J : W \to \mathbb{R}_{>0}$ be a continuous function*[5]*, and let $\mathcal{W}^*$ and J^* be, respectively, the set of the minimizers and the minimum value attained by J over $\mathcal{W}$. For $i \in \{1, \dots, m\}$, assume the following two properties hold true*

$$J(T_i(x)) \le J(x), \qquad \forall\, x \in W, \tag{9}$$
$$J(T_i(x)) < J(x), \qquad \forall\, x \notin \mathcal{W}^*. \tag{10}$$

Consider the trajectory generated by

$$x(t+1) = T_{\sigma(t)}(x(t)), \qquad x(0) \in W,$$

where $\sigma : \mathbb{Z}_{\ge 0} \to \{1, \dots, m\}$[6] *is a process determining which map within the set $\{T_1, \dots, T_m\}$ is selected at iteration t. Then we have*

$$\lim_{t \to \infty} J(x(t)) = J^*$$

and $x(t)$ converges to the set W^*.

Proof: The proof follows using the same continuity arguments adopted in the proof of Theorem 4.3 in [20]. ■

Proof of Theorem IV.2: **Point (i)** is an immediate consequence of the steps of the algorithm. Concerning **Point (ii)**, let us denote $T_{lag}(A_i(t))$ as $T_i(t)$, and let us denote $T_{\max}(t) := \max_i \{T_i(t)\}$. We can state two preliminary facts.

FACT I. *If camera i successfully transmits to camera $i+1$ at time t, and if $T_i(t) > T_{i+1}(t)$, then $T_{i+1}(t+1) < T_i(t)$. As a consequence $T_{i+1}(t') < T_{\max}(t)$, $\forall t' > t$.*

[5] $\mathbb{R}_{>0}$ denotes the set of positive real numbers.
[6] $\mathbb{Z}_{\ge 0}$ denotes the set of non-negative integer numbers.

To confirm the validity of the above fact we observe that, due to the algorithm step, it holds

$$T_{i+1}(t+1) \leq 2\left(\frac{3}{8}T_{i+1}(t) + \frac{T_i(t)}{8}\right) < T_i(t).$$

Since $T_i(t) \leq T_{\max}(t)$, the last sentence follows by induction.

FACT II. *If camera* $i+1$ *successfully transmits to camera* i *at time t, and if* $T_i(t) > T_{i+1}(t)$ *and* $\ell_{i+1}(t) < r_i(t)$*, then* $T_i(t+1) < T_i(t)$.

Since $T_i(t) > T_{i+1}(t)$, the algorithm tries to diminish $T_i(t)$. The fact that $\ell_{i+1}(t) < r_i(t)$ allows to argue that $r_i(t+1) < r_i(t)$. The statement easily follows.

Now, observe that from Fact I it follows that $J_\infty(t)$ is non increasing. To prove that $J_\infty(t+\tau_{\max}) < J_\infty(t)$ if $\xi(t) \notin \mathcal{E}^*_{\mathcal{P}_1}$, we first suppose that only camera i is such that $T_i(t) = T_{\max}(t)$. Since $\xi(t) \notin \mathcal{E}^*_{\mathcal{P}_1}$, it holds that $T_i(t) = T_{\max}(t) > T^*_{\mathcal{P}_1}$. As a consequence it is not possible to have that both, $r_{i-1}(t) = \bar{d}_{i-1}$ and $\ell_{i+1}(t) = \underline{d}_{i+1}$. Suppose also that $\ell_{i+1}(t) = r_i(t)$ and $\ell_{i+1} > \underline{d}_{i+1}$ (all the other starting situations lead to the same conclusion). Due to the assumptions, defining $\tilde{\tau} = h\tau$, there exists a $\tilde{t}$, $t \leq \tilde{t} \leq t+\tilde{\tau}$ such that camera i successfully communicates with camera $i+1$. As a consequence, $\ell_{i+1}(\tilde{t}+1) < r_i(t)$ due to Fact I. If the backward communication works, $T_i(\tilde{t}+1) < T_i(t) = T_{\max}(t)$ and we are done. Otherwise in $[\tilde{t}+1, \tilde{t}+\tilde{\tau}]$ there is a working forward communication between cameras $i+1$ and i, for which (due to Fact I) the hypothesis of Fact II hold. As a consequence, we have that for sure $T_i(t+2\tilde{\tau}+1) < T_i(t) = T_{\max}(t)$.

If there is more than one camera i such that $T_i(t) = T_{\max}(t)$, it is possible to show using the previous reasoning that $J_\infty(t+2\tilde{\tau}(N-1)+1) < J_\infty(t)$. This is the time interval required for the two worst possible cases: one of these is when at time t cameras $1,\dots,N-1$ have time lag $T_{\max}(t)$, are such that $r_i(t) = \ell_{i+1}(t), i = 1\dots,N-1$, and only the last camera has a time lag smaller than $T_{\max}(t)$ (the other case is the one with cameras $2,\dots,N$ that have time lag $T_{\max}(t)$). Defining $\tau_{\max} := 2\tilde{\tau}(N-1)+1$ we are done.

Finally we prove **point (iii)**: consider vector $\xi(t) \in \mathbb{R}^{2N}$ associated to $\{A_i(t)\}_{i=1}^N$, and its reduced version $\xi'(t) \in \mathbb{R}^{2N-2}$ corresponding to $\xi(t)$ without its first and last elements (that are always 0 and L respectively). Consider the sequence $\{x_k\}_{k=1}^\infty$ that represents the evolution of the patrolling areas given by the algorithm every $\tau_{\max}$ instants, i.e. $x_k = \xi'(t)$ for some $t \geq 0$ and $x_{k+1} = \xi'(t+\tau_{\max})$. Due to the physical bounds of the cameras, x_k belongs to a compact set W obtained as the cartesian product of intervals.

Now we can define maps $T_1,\dots,T_M$, with M a finite integer, in the following way: there exists a map $T_j : W \to W$ for every possible camera activation sequence of length $\tau_{\max}-1$ and the related communications that work for each activation, respecting both Assumptions III.1 and IV.1. In this way, it is always possible to find a $j \in \{1,\dots,M\}$ such that $x_{k+1} = T_j(x_k)$. Since each possible step of the algorithm is a continuous function, also every T_i is a continuous function.

Note now that J_∞ is a continuous function such that $J_\infty(T_j(x_k)) \leq J_\infty(x_k),\ \forall x_k \in W, j \in \{1,\dots,M\}$, and $J_\infty(T_j(x_k)) < J_\infty(x_k),\ \forall x_k \notin \mathcal{A}^*_{\mathcal{P}_1}, j \in \{1,\dots,M\}$ due to point (ii). Using Theorem B.1 we have that $J_\infty(x_k)$ converges to J^*_∞. Since at each iteration of the *r-CB* algorithm the cost function is smaller or equal to the previous step, we also have that $J_\infty(t)$ converges to J^*_∞. ■

REFERENCES

[1] B. Song, A. T. Kamal, C. Soto, C. Ding, J. A. Farrell, and A. K. Roy-Chowdhury, "Tracking and activity recognition through consensus in distributed camera networks," *Image Processing, IEEE Transactions on*, vol. 19, no. 10, pp. 2564–2579, 2010.

[2] N. Kariotoglou, D. M. Raimondo, S. J. Summers, and J. Lygeros, "Multi-agent autonomous surveillance: a framework based on stochastic reachability and hierarchical task allocation," *Journal of dynamic systems, measurement, and control*, vol. 137, no. 3, p. 031008, 2015.

[3] C. Ding, B. Song, A. Morye, J. A. Farrell, and A. K. Roy-Chowdhury, "Collaborative sensing in a distributed ptz camera network," *Image Processing, IEEE Transactions on*, vol. 21, no. 7, pp. 3282–3295, 2012.

[4] R. Alberton, R. Carli, A. Cenedese, and L. Schenato, "Multi-agent perimeter patrolling subject to mobility constraints," in *American Control Conference (ACC), 2012*. IEEE, 2012, pp. 4498–4503.

[5] F. Pasqualetti, F. Zanella, J. R. Peters, M. Spindler, R. Carli, and F. Bullo, "Camera network coordination for intruder detection," *Control Systems Technology, IEEE Transactions on*, vol. 22, no. 5, pp. 1669–1683, 2014.

[6] D. Borra, F. Pasqualetti, and F. Bullo, "Continuous graph partitioning for camera network surveillance," *Automatica*, vol. 52, pp. 227–231, 2015.

[7] S. M. Huck, N. Kariotoglou, S. Summers, D. M. Raimondo, and J. Lygeros, "Design of importance-map based randomized patrolling strategies," in *Complexity in Engineering (COMPENG), 2012*. IEEE, 2012, pp. 1–6.

[8] D. M. Raimondo, N. Kariotoglou, S. Summers, and J. Lygeros, "Probabilistic certification of pan-tilt-zoom camera surveillance systems," in *Decision and Control and European Control Conference (CDC-ECC), 2011 50th IEEE Conference on*. IEEE, 2011, pp. 2064–2069.

[9] Y. Chevaleyre, "Theoretical analysis of the multi-agent patrolling problem," sep. 2004, pp. 302 – 308.

[10] H.-N. Chu, A. Glad, O. Simonin, F. Sempé, A. Drogoul, and F. Charpillet, "Swarm approaches for the patrolling problem, information propagation vs. pheromone evaporation," in *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, vol. 1. IEEE, 2007, pp. 442–449.

[11] T. Mao and L. Ray, "Frequency-based patrolling with heterogeneous agents and limited communication," *arXiv preprint arXiv:1402.1757*, 2014.

[12] C. Cassandras, X. Lin, and X. Ding, "An optimal control approach to the multi-agent persistent monitoring problem," *Automatic Control, IEEE Transactions on*, vol. 58, no. 4, pp. 947–961, 2013.

[13] J. J. Acevedo, B. C. Arrue, I. Maza, and A. Ollero, "Cooperative perimeter surveillance with a team of mobile robots under communication constraints," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE, 2013, pp. 5067–5072.

[14] J. J. Acevedo, B. C. Arrue, J. M. Diaz-Bañez, I. Ventura, I. Maza, and A. Ollero, "One-to-one coordination algorithm for decentralized area partition in surveillance missions with a team of aerial robots," *Journal of Intelligent & Robotic Systems*, vol. 74, no. 1-2, pp. 269–285, 2014.

[15] N. H. Vaidya, C. N. Hadjicostis, and A. D. Dominguez-Garcia, "Distributed algorithms for consensus and coordination in the presence of packet-dropping communication links-part ii: Coefficients of ergodicity analysis approach," *arXiv preprint arXiv:1109.6392*, 2011.

[16] G. Belgioioso, A. Cenedese, and G. Michieletto, "Distributed partitioning strategies with visual optimization for camera network perimeter patrolling," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, Dec 2016, pp. 5912–5917.

[17] J. Czyzowicz, L. Gasieniec, A. Kosowski, and E. Kranakis, "Boundary patrolling by mobile agents with distinct maximal speeds," in *Algorithms–ESA 2011*. Springer, 2011, pp. 701–712.

[18] A. Kawamura and Y. Kobayashi, "Fence patrolling by mobile agents with distinct speeds," *Distributed Computing*, vol. 28, no. 2, pp. 147–154, 2015.

[19] G. Notarstefano and F. Bullo, "Distributed abstract optimization via constraints consensus: Theory and applications," *Automatic Control, IEEE Transactions on*, vol. 56, no. 10, pp. 2247–2261, 2011.

[20] F. Bullo, R. Carli, and P. Frasca, "Gossip coverage control for robotic networks: Dynamical systems on the space of partitions," *SIAM Journal on Control and Optimization*, vol. 50, no. 1, pp. 419–447, 2012.