Preprints,18th IFAC Symposium on System Identification
July 9-11, 2018. Stockholm, Sweden

Multi-Sensor Task Assignment Using Linear Matrix Inequalities in the Multiple Target Tracking Problem ⋆

Victoria Erofeeva, * **Oleg Granichin,** ** **Olga Granichina,** ***

* *Saint Petersburg State University (Faculty of Mathematics and Mechanics), Institute of Problems in Mechanical Engineering, Russian Academy of Sciences, St. Petersburg, Russia*
e-mail: victoria@grenka.net
** *Saint Petersburg State University (Faculty of Mathematics and Mechanics, and Research Laboratory for Analysis and Modeling of Social Processes), Institute of Problems in Mechanical Engineering, Russian Academy of Sciences, St. Petersburg, Russia*
e-mail: o.granichin@spbu.ru
*** *Herzen State Pedagogical University, St. Petersburg, Russia*
e-mail: olga_granichina@mail.ru

Abstract: Due to significant advancements in embedded systems, sensor devices and wireless communication technology, sensor networks have been attracting widespread attention in areas such as target tracking, monitoring, and surveillance. Technological advancements made it possible to deploy a large number of inexpensive but technically advanced sensors to cover wide areas. However, when a tracking system has to track a large number of targets, the computation and communication loads arise. In this paper we propose a task assignment algorithm based on linear matrix inequalities (LMI) to reduce the computational complexity and communication load. Simulation results and a comparison with the Kalman filtering strategy confirm the suitability of the approach.

Keywords: sensor network, multiple target tracking, linear matrix inequalities, ellipsoidal approximation, task assignments

1. INTRODUCTION

Due to significant advancements in embedded systems, sensor devices and wireless communication technology, sensor networks have been attracting widespread attention in areas such as target tracking, monitoring, and surveillance. Technological advancements made it possible to deploy a large number of inexpensive but technically advanced sensors to cover wide areas (Dargie and Poellabauer (2010)). Applications in these fields include, for example, intelligent video surveillance at cluttered and crowded places, air traffic control, space situational awareness and animal tracking (see Hanif et al. (2017); Thite and Mishra (2016); Jia et al. (2016)). Deployment of multiple sensors provides more advantages over a single node. In particular, each sensor mostly receives incomplete observations (measurements) because of the noisiness of an environment and inaccuracy inherent to the sensor devices. Thanks to the use of multiple sensors one might obtain more accurate estimation of the measured value through the information fusion. In other words, multi-sensor networks can be used to reduce uncertainties.

Sensor networks that contain multiple nodes with sensing, processing and communication capabilities are ubiquitous in tracking systems (Wei et al. (2016)). In general, sensor networks perform estimates of some state of a dynamic process through communication between the network nodes. Currently, three communication schemes are used for multi-sensor networks: centralized, distributed and hybrid. Centralized systems have the most accurate estimation since they are aware of all or almost all measurements obtained by the sensors. In turn, distributed systems have such properties as robustness to failures and scalability as well as less throughput requirements compared to centralized ones. Hybrid systems provide a trade-off between the properties of centralized and distributed systems. In such systems one may combine a distributed control strategy with local fusion centers.

The use of the tracking systems, which are comprised of multiple inexpensive and small sensors, brings new challenges due to resource limitations of the network. Each sensor has limited sensing coverage and it might be ineffective for a target to be tracked by all available sensors or by a fixed subset of sensors through the entire tracking process. Moreover, sensors deployed in a large area of interest may not contribute much to the tracking quality since sensors might be far away from the targets. Nevertheless, they consume their own and network resources collecting the data and communicating with the other nodes. These issues gave rise to the sensor selection problem, in which

⋆ This work was supported by Russian Science Foundation (project 16-19-00057).

the best subset of the available sensors needs to be chosen according to given performance constraints.

In general, the sensor selection problem is expressed as follows (Chepuri and Leus (2015)):

$$\arg\min_{\mathbf{q}\in\{0,1\}^N} h(\mathbf{Q}(\mathbf{q})) \;\; s.t. \;\; \mathbf{1}_N^{\mathrm{T}}\mathbf{q} = K, \tag{1}$$

where $\mathbf{q}$ is a selection vector of length N, $h(\mathbf{Q}(\mathbf{q}))$ is a scalar cost function related to the error covariance matrix $\mathbf{Q}$. The error covariance matrix is optimized to select the best subset of K sensors out of N available sensors. The problem (1) is combinatorial and one needs to do $\binom{K}{N}$ searches to find the solution. In the multi-target case this problem becomes even worse because we need to find a selection vector for every target.

1.1 Related Works

There are various algorithms available in literature to solve the problem of sensor selection or resource allocation. In Masazade et al. (2012) the authors propose a target tracking algorithm based on extended Kalman filtering, in which the selection process is performed by designing a sparse gain matrix. Heuristic Yu and Prasanna (2005), stochastic heuristic Jin et al. (2012) and meta-heuristic intelligent optimization algorithms Yang et al. (2014) have also been considered to solve this problem. Exhaustive search Kaplan (2006), greedy search Kalandros (2002), auction algorithm Chen et al. (2006), are some other algorithms that are applied to resource allocation. Considered algorithms tend to be computationally expensive. To address the complexity issues, sparse convex optimization approaches are used in Joshi and Boyd (2009).

In Botts et al. (2016) the authors consider a stochastic multi-agent and multi-target surveillance problem and apply to it a cyclic stochastic optimization algorithm. Recently, researchers have been actively developing approaches based on randomization (see Tempo et al. (2012); Granichin et al. (2015)). Event-based tracking technologies are also widely considered due to throughput constraints and difficulty in analyzing a large amount of data Batmani et al. (2017).

The existing works mainly address the problem of choosing K sensors from a set of available sensors in order to obtain the best tracking accuracy. However, in large-scale networks it is important to find a trade-off between accuracy and resource utilization.

1.2 Contribution

In this work we suggest to use of a hybrid system scheme and propose a task assignment algorithm based on linear matrix inequalities (LMI) to deal with the computational complexity problem. The selection problem is formulated as the design of a sparse resource allocation matrix $\mathbf{G}_t$ to choose the most informative sensors. The entries of $\mathbf{G}_t$ are designed to be as sparse as possible such that the tracking error and the amount of used sensors are minimized.

Mathematically, it means we should minimize the sum of non-zero entries of the vector $\mathbf{g}$ defined by the l_0-(quasi) norm: $\|\mathbf{g}\|_0 = \sum_{j=1}^{N} |\mathrm{sign}\, g_j|$. Since the l_0-(quasi) norm optimization is NP-hard and nonconvex, one should use the convex surrogate, i.e. the l_1-norm heuristic, that gives the best approximation of the sparse solution (Barabanov and Granichin (1984); Polyak et al. (2014)):

$$\|\mathbf{g}\|_1 = \sum_{j=1}^{N} |g_j|.$$

In essence, we seek a sparse matrix $\mathbf{G}_t$ consisting of vectors $\mathbf{g}$ (i.e., vector with many zeros and a few non-zero entries) that minimizes the quality functional presented in the next section.

1.3 Outline and Notations

The remainder of the paper is organized as follows. In Section 2, we introduce the problem of multiple target tracking by a sensor network, consisting of identical devices. Section 3 provides the technique of finding an intersection of the ellipsoids corresponding to the sensors measurements. In Section 4, we propose an algorithm based on LMI approach to find sensor subsets. Finally, in Section 5 a simulation experiment is provided.

The notation used in this paper can be described as follows. Upper and lower bold face letters are used for matrices and column vectors, respectively. $\mathrm{E}\{\cdot\}$ is the expectation operation. $\mathbf{I}_k$ is a $k \times k$ identity matrix with ones on the main diagonal and zeros elsewhere. $\preccurlyeq$ is a non-strict inequality for symmetric matrices that is understood in the sense of inequalities for quadratic forms. $(\cdot)^{\mathrm{T}}$ denotes transposition. $|\mathcal{U}|$ denotes the cardinality of the set $\mathcal{U}$. $\|\cdot\|$ is the Euclidean norm. $\mathrm{tr}\{\cdot\}$ is the matrix trace operator. $\det\{\cdot\}$ is the matrix determinant.

2. PROBLEM STATEMENT

Consider a distributed network of n sensors, randomly located in an area of interest. Let $N = \{1, 2, \ldots, n\}$ be the set of sensors and $\mathbf{s}_t^j \in \mathbb{R}^d$ be the state of the sensor j. In the line of sight of the sensors are moving m targets. Our goal is to assign sensors to the targets in such a way that we could accurately predict the movement trajectories of the targets and use as less sensors as possible.

Let $M = \{1, 2, \ldots, m\}$ be the set of targets, $\{\mathbf{r}_t^i\}_{t=0,1,2,\ldots}$, $\mathbf{r}_t^i \in \mathbb{R}^p$, $i \in M$ be the movement trajectory of the target i, whose state changes according to the following equation:

$$\mathbf{r}_{t+1}^i = f^i(\mathbf{r}_t^i) + \mathbf{w}_t^i, \tag{2}$$

where $f^i(\cdot)$ is a state-transition function, $\{\mathbf{w}_t^i\}$ is the white Gaussian noise with zero mathematical expectation and covariance matrix $\mathbf{R}_w^i$: $\mathrm{E}\mathbf{w}_t^i = 0$, $\mathrm{E}\mathbf{w}_t^i(\mathbf{w}_t^i)^{\mathrm{T}} = \mathbf{R}_w^i \preccurlyeq \sigma_w^2\mathbf{I}_k$.

The sensors estimate the state $\mathbf{r}_t^i$ of the object i based on measurements received in accordance with the following observation model

$$\mathbf{z}_t^{i,j} = \varphi(\mathbf{s}_t^j, \mathbf{r}_t^i) + \boldsymbol{\varepsilon}_t^{i,j}, \tag{3}$$

where $\mathbf{z}_t^{i,j} \in \mathbb{R}^q$ is a measurement of the state of the object i available to the sensor j at time instant t, $\varphi(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}^q$ is an observation function, which depends on the current state of the object i and sensor j, $\{\boldsymbol{\varepsilon}_t^{i,j}\}$ is the additive external noise with zero mean $\mathrm{E}\boldsymbol{\varepsilon}_t^{i,j} = 0$ and the error covariance matrix $\mathrm{E}\boldsymbol{\varepsilon}_t^{i,j}(\boldsymbol{\varepsilon}_t^{i,j})^{\mathrm{T}} = \Sigma_t^{i,j}$.

We assume that for any $i \in M$, $j \in N$ and independent centered $\varepsilon_t^{i,j}$ with the error covariance matrix $\Sigma_t^{i,j}$ there exists an inverse function $\varphi^{-1}(\mathbf{s}_t^j, \cdot) : \mathbb{R}^q \to \mathbb{R}^p$:

$$\varphi^{-1}(\mathbf{s}_t^j, \varphi(\mathbf{s}_t^j, \mathbf{r}_t^i) + \varepsilon_t^{i,j}) = \mathbf{r}_t^i + \boldsymbol{\xi}_t^{i,j}, \tag{4}$$

where $\boldsymbol{\xi}_t^{i,j}$ is an independent component with zero mean $E\boldsymbol{\xi}_t^{i,j} = 0$, the error covariance matrix $\mathrm{E}\boldsymbol{\xi}_t^{i,j}(\boldsymbol{\xi}_t^{i,j})^{\mathrm{T}} = \Xi_t^{i,j}$ and the bounded fourth central moment $\mathrm{E}\|\boldsymbol{\xi}_t^{i,j}\|^4 \leq M_4$.

2.1 Confidence Region

Let a confidence region be represented as an ellipsoid around the point $\boldsymbol{\eta}_t^{i,j} = \varphi^{-1}(\mathbf{s}_t^j, \mathbf{z}_t^{i,j})$. The confidence region with the user-defined significance level p would include the point representing the "true" value $\mathbf{r}_t^i$ with the probability $1 - p$. We define this ellipsoid as follows:

$$\mathcal{E}_t^{i,j} = \{\mathbf{r}_t^i \;:\; (\mathbf{r}_t^i - \boldsymbol{\eta}_t^{i,j})^{\mathrm{T}}(\Xi_t^{i,j})^{-1}(\mathbf{r}_t^i - \boldsymbol{\eta}_t^{i,j}) \leq \chi^2_{p,d}\}, \tag{5}$$

where $\chi^2_{p,d}$ is the p-value matching to the χ^2 distribution for d degrees of freedom.

Example 1. Let one specify an ellipsoid around the point $\boldsymbol{\eta}_t^{i,j} \in \mathbb{R}^2$, i.e. $d = 2$. Let this ellipsoid include the value $\mathbf{r}_t^i$ with the 95% probability. In accordance with the table of χ^2 values vs p-values, one should set the p-value to 0.05.

For each target i we have a set of points $\boldsymbol{\eta}_t^i = \{\boldsymbol{\eta}_t^{i,1}, \ldots, \boldsymbol{\eta}_t^{i,n}\}$ and corresponding to them ellipsoids $\mathcal{E}_t^i = \{\mathcal{E}_t^{i,1}, \ldots, \mathcal{E}_t^{i,n}\}$. We assume that the "true" value $\mathbf{r}_t^i$ belongs to the intersection of the ellipsoids contained in $\mathcal{E}_t^i$ and we would like to find this intersection region. However, it is hard to achieve if the value n is large enough. In this case the intersection region becomes too complex. In order to find it, we are going to adopt the technique of linear matrix inequalities described in the next section.

2.2 Quality Function

We denote by $\boldsymbol{\theta}_t = col(\mathbf{r}_t^1, \ldots, \mathbf{r}_t^m)$ the joint vector of all target states. Let $\widehat{\mathbf{r}}_t^i$ be an estimate of the state of target i at time instant t and $\widehat{\boldsymbol{\theta}}_t = col(\widehat{\mathbf{r}}_t^1, \ldots, \widehat{\mathbf{r}}_t^m)$ be the joint vector of all estimates.

In general, the main goal of the tracking process can be achieved by minimizing the following quality function:

$$\bar{F}_t(\widehat{\boldsymbol{\theta}}_t) = \frac{1}{2}\sum_{i \in M} \|\mathbf{r}_t^i - \widehat{\mathbf{r}}_t^i\|^2 \to \min_{\widehat{\boldsymbol{\theta}}_t}. \tag{6}$$

Let $\hat{\mathcal{E}}_t = \{\hat{\mathcal{E}}_t^1, \ldots, \hat{\mathcal{E}}_t^m\}$ be the set of ellipsoids that approximate the intersections of ellipsoids contained in $\{\mathcal{E}_t^i\}_{i \in M}$. Equivalently, the problem (6) may be represented as follows:

$$\Phi_t(\hat{\mathcal{E}}_t) = \sum_{i \in M} \mathrm{vol}(\hat{\mathcal{E}}_t^i) \to \min_{\hat{\mathcal{E}}_t}, \tag{7}$$

where $\mathrm{vol}(\cdot)$ is the volume.

In order to reduce the processing and communications loads, we are also going to minimize the number of selected sensors. We denote by $\mathbf{G}_t$ the resource allocation matrix that needs to be as sparse as possible. The entities $g_t^{i,j}$ of this matrix indicate whether the sensor j is assigned to the target i or not. Lastly, our quality function takes the following form:

$$\bar{\Phi}_t(\mathbf{G}_t) = \Phi_t(\hat{\mathcal{E}}) + \sum_{i \in M} \|\mathbf{G}_t^{(i,\cdot)}\|_1 \to \min_{\mathbf{G}_t}, \tag{8}$$

where $\mathbf{G}_t^{(i,\cdot)}$ is the i-th row of the matrix $\mathbf{G}_t$.

3. AN INTERSECTION REGION OF ELLIPSOIDS

In Subsection 2.1, we mentioned the problem of finding the intersection region of ellipsoids. In this section we provide an explanation of how we can do that. Let the fusion center receive a set of points $\boldsymbol{\eta}_t^i = \{\boldsymbol{\eta}_t^{i,1}, \ldots, \boldsymbol{\eta}_t^{i,n}\}$ of the i-th target at time instant t. In Boyd et al. (1994) there are several methods that approximate the intersection region of ellipsoids. We are going to use outer approximation to find an ellipsoid $\hat{\mathcal{E}}_t^i$ such that

$$\hat{\mathcal{E}}_t^i \supseteq \bigcap_{j=1}^{n} \mathcal{E}_t^{i,j}. \tag{9}$$

For this purpose we apply the S-procedure, which could be used to obtain a linear matrix inequality (LMI) that is sufficient for (9) to hold. Before the S-procedure application, we should convert the ellipsoid (5) into the following form:

$$\mathcal{E}_t^{i,j} = \{\mathbf{x} \mid H^{i,j}(\mathbf{x}) \leq 0\},$$
$$H^{i,j}(\mathbf{x}) = \mathbf{x}^{\mathrm{T}}\mathbf{A}_t^{i,j}\mathbf{x} + 2\mathbf{x}^{\mathrm{T}}\mathbf{b}_t^{i,j} + \mathbf{c}_t^{i,j}, \tag{10}$$

where $\mathbf{A}_t^{i,j} = (\Xi_t^{i,j})^{-1}$, $\mathbf{b}_t^{i,j} = -(\Xi_t^{i,j})^{-1}\boldsymbol{\eta}_t^{i,j}$, $\mathbf{c}_t^{i,j} = (\boldsymbol{\eta}_t^{i,j})^{\mathrm{T}}(\Xi_t^{i,j})^{-1}\boldsymbol{\eta}_t^{i,j} - 1$.

Each of these forms of representing an ellipsoid can be afterwards converted into each other. In this paper we are considering a special case of an ellipsoid, which is referred to as an ellipse, i.e. when a 2-D plane is considered. Nevertheless, the approach we are going to use is suitable for ellipsoids in higher dimensions as well.

From the S-procedure the following condition could be obtained: there exist positive scalars $\tau^{i,1}, \ldots, \tau^{i,n}$ such that

$$\begin{bmatrix} \hat{\mathbf{A}}^i & \hat{\mathbf{b}}^i \\ (\hat{\mathbf{b}}^i)^{\mathrm{T}} & (\hat{\mathbf{b}}^i)^{\mathrm{T}}(\hat{\mathbf{A}}^i)^{-1}\hat{\mathbf{b}}^i - 1 \end{bmatrix} - \sum_{j=1}^{n} \tau^{i,j} \begin{bmatrix} \mathbf{A}^{i,j} & \mathbf{b}^{i,j} \\ (\mathbf{b}^{i,j})^{\mathrm{T}} & \mathbf{c}^{i,j} \end{bmatrix} \leq 0, \tag{11}$$

which can be written as the LMI (in variables $\hat{\mathbf{A}}^i$, $\hat{\mathbf{b}}^i$, $\hat{\mathbf{c}}^i = (\hat{\mathbf{b}}^i)^{\mathrm{T}}(\hat{\mathbf{A}}^i)^{-1}\hat{\mathbf{b}}^i - 1$, and $\tau^{i,1}, \ldots, \tau^{i,n}$). Finally, we will be able to find the ellipsoid $\hat{\mathcal{E}}_t^i$, which has the smallest volume, by solving the following convex problem:

$$\begin{gathered} \text{minimize} \quad \log\det(\hat{\mathbf{A}}^i)^{-1} \\ s.\ t.\ \ \hat{\mathbf{A}}^i > 0, \ \ \tau^{i,1} \geq 0, \ldots, \tau^{i,n} \geq 0, \\ \begin{bmatrix} \hat{\mathbf{A}}^i & \hat{\mathbf{b}}^i & 0 \\ (\hat{\mathbf{b}}^i)^{\mathrm{T}} & -1 & (\hat{\mathbf{b}}^i)^{\mathrm{T}} \\ 0 & \hat{\mathbf{b}}^i & -\hat{\mathbf{A}}^i \end{bmatrix} - \sum_{j=1}^{n} \tau^{i,j} \begin{bmatrix} \mathbf{A}^{i,j} & \mathbf{b}^{i,j} & 0 \\ (\mathbf{b}^{i,j})^{\mathrm{T}} & \mathbf{c}^{i,j} & 0 \\ 0 & 0 & 0 \end{bmatrix} \leq 0. \end{gathered} \tag{12}$$

Listing 1 demonstrates the example of a procedure that finds outer approximation of the intersection of ellipsoids. In the code below we utilize the CVX library (see Grant et al. (2008)).

Listing 1. Outer approximation of the intersection of ellipsoids in Matlab

```
cvx_begin sdp
  variable A(n,n) symmetric;
  variable b(n,1);
  variable tau(1,p);

  minimize(-log_det(A));
  subject to
  A > 0;
  hat_ellipse = zeros(2*n+1, 2*n+1);
  for i = 1:p
    tau(i) ≥ 0;
    hat_ellipse = hat_ellipse + tau(i) * ...
        [ellipses(i).A ellipses(i).b ...
        zeros(n, n); ellipses(i).b' ...
        ellipses(i).c zeros(1, n); zeros(n, ...
        n) zeros(n, n) zeros(n, 1)];
  end
  [A b zeros(n, n); b' -1 b'; zeros(n, n) b ...
      -A] - hat_ellipse ≤ 0;
cvx_end
```

In the next section we describe how to solve problem (8) using linear matrix inequalities.

4. LMI-BASED SOLUTION

In order to solve problem (8) we need to modify (12) in such a way that the method takes into account the resource allocation matrix $\mathbf{G}_t$. We made a slight change in the problem (12), adding the new conditions as follows:

$$\begin{gathered} \text{minimize} \quad \delta \\ s.\,t.\ \ \forall i \quad \hat{\mathbf{A}} > 0, \quad g^{i,1} \geq 0, \ldots, g^{i,n} \geq 0, \qquad (13) \\ \begin{bmatrix} \hat{\mathbf{A}}^i & \hat{\mathbf{b}}^i & 0 \\ (\hat{\mathbf{b}}^i)^{\mathrm{T}} & -1 & (\hat{\mathbf{b}}^i)^{\mathrm{T}} \\ 0 & \hat{\mathbf{b}}^i & -\hat{\mathbf{A}}^i \end{bmatrix} - \sum_{j=1}^{n} g^{i,j} \begin{bmatrix} \mathbf{A}^{i,j} & \mathbf{b}^{i,j} & 0 \\ (\mathbf{b}^{i,j})^{\mathrm{T}} & \mathbf{c}^{i,j} & 0 \\ 0 & 0 & 0 \end{bmatrix} \leq 0. \\ \sum_{i=1}^{m} \log\det(\hat{\mathbf{A}}^i)^{-1} + \alpha \sum_{i=1}^{m} \|\mathbf{G}_t^{(i,\cdot)}\|_1 \leq \delta, \end{gathered}$$

where α is the regularization coefficient.

The problem (13) means that we would like to find an ellipsoid with the volume as small as possible while using as few sensors as possible. In that case, we agree to get an estimation with some quality loss, but instead we reduce computational and communication loads.

Let us consider two examples that show the difference between (12) and (13), see Fig. 1-2. We set $m = 1$ and $n = 3$, i.e. one target and three sensors. In Fig. 1 we use α equal to 1 and get an ellipse with the volume equal to 0.1715 and $\boldsymbol{\tau}^1 = [0.4122, 0.5225, 0.1896]^{\mathrm{T}}$. In turn, in figure 2 we use α equal to 1 and get an ellipse with the volume equal to 0.1865 and $\mathbf{g}^1 = [0.4827, 0.5173, 0.0000]^{\mathrm{T}}$. In other words, in the first case we calculate the estimate using all sensors and in the second one using all except the last one.

In real applications there may be some restrictions regarding the maximum number of targets that can be tracked by each sensor, i.e. $|G_t^j| \leq g_{\max}^j$. The solution of (13) does not guarantee that this restriction will hold. It only minimizes the value of $|G_t^j|$. To deal with this issue one may use the tracking algorithm that holds this restriction, like the parameter estimation method presented in Granichin and Erofeeva (2018).

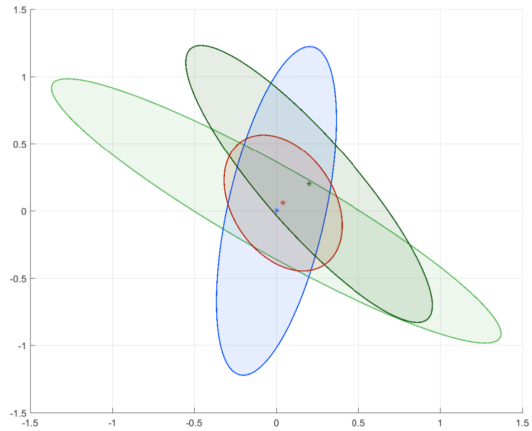

Fig. 1. Outer approximation based on all measurments

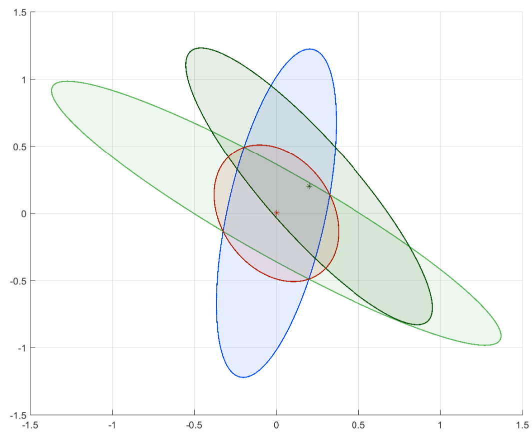

Fig. 2. Outer approximation based on some part of measurment set

Now, we formulate the tracking algorithm with LMI based task assignment. Consider a hybrid network scheme, which consists of a fusion center and a sensor network. We assume that the solution of (13) is suboptimal during the time period T.

Algorithm 1. Task assignment

Input: $\alpha > 0$, $\boldsymbol{\eta}_t^i = \{\boldsymbol{\eta}_t^{i,1}, \ldots, \boldsymbol{\eta}_t^{i,n}\}$, $i \in M$

Output: $\mathbf{G}_t$

1. Initialization. Associate each point $\boldsymbol{\eta}_t^{i,j}$ with the ellipsoid $\mathcal{E}_t^{i,j}$.

2. Convert the ellipsoids to the form (10).

3. Solve the problem (13).

4. If the problem (13) is feasible. For $i \leftarrow 1$ to m:

4.1 For $j \leftarrow 1$ to n:

4.1.1 If $g^{i,j} > 0$: assign the sensor j to the target i.

5. For each sensor j form the neighbors set N_t^j, i.e. sensors that track the same targets. Send assignments to the sensors.

Algorithm 2. Tracking process on each sensor

Input: the neighbors set N_t^j

Output: $\widehat{\boldsymbol{\theta}}_t^j$

1. Initialization. Set counter $n = 0$.

2. Iteration $n \to n + 1$.

3. If $n \,\%\, T == 0$:

3.1 Send own measurements to the fusion center.

4. Perform the estimation process based on a predefined algorithm (e.g. Cyclic Simultaneous Perturbation Stochastic Approximation Granichin and Erofeeva (2018)).

5. Read the message queue: if there is a message from the fusion sensor go to step 5.1; else go to step 2.

5.1 Update the neighbor set and assignments.

5.2 Go to step 2.

5. SIMULATION RESULTS

5.1 Observation Model

We consider a 2D-plane, in which the state of the target i is $\mathbf{r}_t^i = [r_t^{i,1}\ r_t^{i,2}\ \dot{r}_t^{i,1}\ \dot{r}_t^{i,2}]^{\mathrm{T}}$ and the state of the sensor j is $\mathbf{s}_t^j = [s_t^{j,1}\ s_t^{j,2}\ \dot{s}_t^{j,1}\ \dot{s}_t^{j,2}]^{\mathrm{T}}$. Each state consists of position and velocity components at time instant t. Suppose the sensors are able to determine the angle and distance to the objects, then:

$$\varphi(\mathbf{s}_t^j, \mathbf{r}_t^i) = \begin{bmatrix} \psi(\mathbf{s}_t^j, \mathbf{r}_t^i) \\ \rho(\mathbf{s}_t^j, \mathbf{r}_t^i) \end{bmatrix} \in \mathbb{R}^2, \tag{14}$$

where

$$\psi(\mathbf{s}_t^j, \mathbf{r}_t^i) = \operatorname{arctg}\left[\frac{r_t^{i,1} - s_t^{j,1}}{r_t^{i,2} - s_t^{j,2}}\right] \tag{15}$$

is the angle to the object i,

$$\rho(\mathbf{s}_t^j, \mathbf{r}_t^i) = \sqrt{\left(r_t^{i,1} - s_t^{j,1}\right)^2 + \left(r_t^{i,2} - s_t^{j,2}\right)^2} \tag{16}$$

is the distance to the object i.

In this case, the inverse function $\varphi^{-1}(\mathbf{s}_t^j, \cdot)$ is as follows

$$\varphi^{-1}(\mathbf{s}_t^j, \mathbf{z}_t^{i,j}) = \mathbf{s}_t^j + \begin{bmatrix} z_t^{i,j,2} \sin z_t^{i,j,1} \\ z_t^{i,j,2} \cos z_t^{i,j,1} \end{bmatrix}, \tag{17}$$

where $z_t^{i,j,1}$ and $z_t^{i,j,2}$ are the first and second coordinates of the vector $\mathbf{z}_t^{i,j}$, respectively. If the covariance matrices $\boldsymbol{\varepsilon}_t^{i,j}$ are equal to $\Sigma_t^{i,j} = \begin{bmatrix} \sigma_\psi^2 & 0 \\ 0 & (z_t^{i,j,2}\sigma_\rho)^2 \end{bmatrix}$, then the covariance of $\boldsymbol{\xi}_t^{i,j}$ is

$$\Xi_t^{i,j} = R(z_t^{i,j,1}) \begin{bmatrix} (z_t^{i,j,2}\sigma_\psi)^2 & 0 \\ 0 & (z_t^{i,j,2}\sigma_\rho)^2 \end{bmatrix} R(z_t^{i,j,1})^{\mathrm{T}}, \tag{18}$$

where $R(\psi) = \begin{bmatrix} \sin\psi & -\cos\psi \\ \cos\psi & \sin\psi \end{bmatrix}$ is the rotation matrix through the angle ψ.

5.2 Experiments

Next, we consider one possible experiment setting. Six targets move uniformly and rectilinearly in a square area of interest with identical and constant velocities. The area is of size $300 \times 300\ km^2$ and velocities $\dot{r}_t^{i,1}$ and $\dot{r}_t^{i,2}$ are equal to 2500 km/h. We define the state-transition function as follows

$$f(\mathbf{r}_t^i) = \mathbf{D}^i \mathbf{r}_t^i; \quad \mathbf{D}^i = \begin{pmatrix} 1 & \delta t & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \delta t \\ 0 & 0 & 0 & 1 \end{pmatrix}; \quad \delta t = 1\ sec.$$

In the area of interest we randomly locate six sensors. The noise in the measurements obtained by each sensor is set to the following values:

$$\Sigma_t^{i,j} = \begin{bmatrix} \sigma_\psi^2 & 0 \\ 0 & (z_t^{i,j,2}\sigma_\rho)^2 \end{bmatrix} = \begin{bmatrix} 0.7^2 & 0 \\ 0 & \left(\dfrac{z_t^{i,j,2}}{100}\right)^2 \end{bmatrix}$$

The targets begin their movement starting at the initial positions, which are defined as $\mathbf{r}_0^1 = [270, 295]$; $\mathbf{r}_0^2 = [240, 290]$; $\mathbf{r}_0^3 = [210, 285]$; $\mathbf{r}_0^4 = [180, 280]$; $\mathbf{r}_0^5 = [150, 275]$; $\mathbf{r}_0^6 = [120, 270]$. We assume that the sensor network is homogeneous, i.e. the characteristics of each sensor are the same. By characteristics we mean, for example, the field of view, which is assumed to be 360 degrees. The field of view also covers the whole considered area of interest. The duration of experiments is 200 iterations.

For simplicity, instead of visualizing all targets we show only one target with id #1 in figures 3 and 4. We use the following notation for the figures: the dotted line is the trajectory of a target's movement; the solid line with the marker "square" is an estimation of the trajectory; the circle is a sensor.

Figure 3 demonstrates the estimation of the movement trajectory with task assignment process based on LMI approach discussed above. In turn, for comparison we provide the estimation based on Kalman Filter performed on the fusion center, see Fig. 4. The main difference between these two cases is that in the second case the measurements from all sensors were used. In contrast to this case, LMI-based solution was obtained with the use of formed sensor subsets only.

REFERENCES

Barabanov, A. and Granichin, O. (1984). Optimal controller for linear plants with bounded noise. *Automation and Remote Control*, 45(5), 39–46.

Batmani, Y., Davoodi, M., and Meskin, N. (2017). Event-triggered suboptimal tracking controller design for a class of nonlinear discrete-time systems. *IEEE Transactions on Industrial Electronics*.

Botts, C.H., Spall, J.C., and Newman, A.J. (2016). Multi-agent surveillance and tracking using cyclic stochastic gradient. In *American Control Conference (ACC), 2016*, 270–275. IEEE.

Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V. (1994). *Linear matrix inequalities in system and control theory.* SIAM.

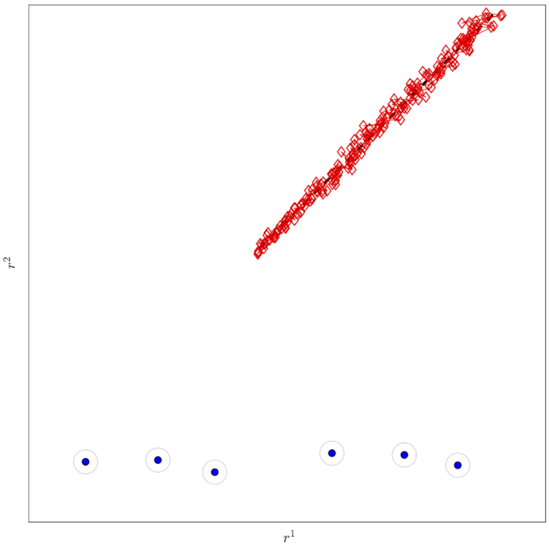

Fig. 3. The estimation with LMI based task assignment

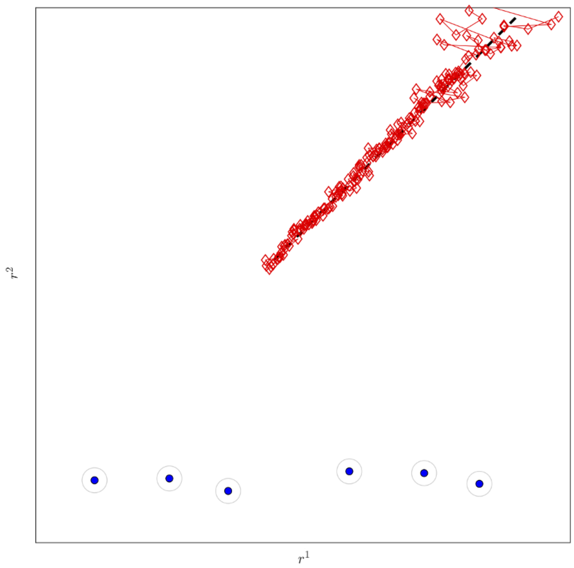

Fig. 4. The estimation based on Kalman filter

Chen, J., Zang, C., Liang, W., and Yu, H. (2006). Auction-based dynamic coalition for single target tracking in wireless sensor networks. In *Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on*, volume 1, 94–98. IEEE.

Chepuri, S.P. and Leus, G. (2015). Sparsity-promoting sensor selection for non-linear measurement models. *IEEE Transactions on Signal Processing*, 63(3), 684–698.

Dargie, W. and Poellabauer, C. (2010). *Fundamentals of wireless sensor networks: theory and practice*. John Wiley & Sons.

Granichin, O. and Erofeeva, V. (2018). Cyclic stochastic approximation with disturbance on input in the parameter tracking problem based on a multiagent algorithm. *Automation and Remote Control*, 79(6), 996–1011.

Granichin, O., Volkovich, Z., and Toledano-Kitai, D. (2015). Randomized algorithms in automatic control and data mining.

Grant, M., Boyd, S., and Ye, Y. (2008). Cvx: Matlab software for disciplined convex programming.

Hanif, A., Mansoor, A.B., and Imran, A.S. (2017). Deep multi-view correspondence for identity-aware multi-target tracking. In *2017 International Conference on Digital Image Computing: Techniques and Applications (DICTA)*, 1–8. doi:10.1109/DICTA.2017.8227423.

Jia, B., Pham, K.D., Blasch, E., Shen, D., Wang, Z., and Chen, G. (2016). Cooperative space object tracking using space-based optical sensors via consensus-based filters. *IEEE Transactions on Aerospace and Electronic Systems*, 52(4), 1908–1936.

Jin, Y., Jin, J., Gluhak, A., Moessner, K., and Palaniswami, M. (2012). An intelligent task allocation scheme for multihop wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 23(3), 444–451.

Joshi, S. and Boyd, S. (2009). Sensor selection via convex optimization. *IEEE Transactions on Signal Processing*, 57(2), 451–462.

Kalandros, M. (2002). Covariance control for multisensor systems. *IEEE Transactions on Aerospace and Electronic Systems*, 38(4), 1138–1157.

Kaplan, L.M. (2006). Global node selection for localization in a distributed sensor network. *IEEE Transactions on Aerospace and Electronic systems*, 42(1), 113–135.

Masazade, E., Fardad, M., and Varshney, P.K. (2012). Sparsity-promoting extended kalman filtering for target tracking in wireless sensor networks. *IEEE Signal Processing Letters*, 19(12), 845–848.

Polyak, B.T., Khlebnikov, M.V., and Shcherbakov, P.S. (2014). Sparse feedback in linear control systems. *Automation and Remote Control*, 75(12), 2099–2111.

Tempo, R., Calafiore, G., and Dabbene, F. (2012). *Randomized algorithms for analysis and control of uncertain systems: with applications*. Springer Science & Business Media.

Thite, A. and Mishra, A. (2016). Optimized multi-sensor multi-target tracking algorithm for air surveillance system. In *Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 2016 2nd International Conference on*, 637–642. IEEE.

Wei, B., Nener, B., Liu, W., and Ma, L. (2016). Centralized multi-sensor multi-target tracking with labeled random finite sets. In *Control, Automation and Information Sciences (ICCAIS), 2016 International Conference on*, 82–87. IEEE.

Yang, J., Zhang, H., Ling, Y., Pan, C., and Sun, W. (2014). Task allocation for wireless sensor network using modified binary particle swarm optimization. *IEEE Sensors Journal*, 14(3), 882–892.

Yu, Y. and Prasanna, V.K. (2005). Energy-balanced task allocation for collaborative processing in wireless sensor networks. *Mobile Networks and Applications*, 10(1-2), 115–131.