Les Cahiers du GERAD ISSN: 0711–2440

Filtering for Detecting Multiple Targets Trajectories on a One-Dimensional Torus

Ivan Gentil
Bruno Rémillard

G–2003–09

February 2003

Filtering for Detecting Multiple Targets Trajectories on a One-Dimensional Torus

Ivan Gentil

Bruno Rémillard*

Service de l'enseignement des méthodes quantitatives de gestion
HEC Montréal
3000, chemin de la côte Sainte-Catherine
Montréal (Québec) Canada H3T 2A7
**and GERAD*
{ivan.gentil,bruno.remillard}@hec.ca

February, 2003

Les Cahiers du GERAD

G–2003–09

Abstract

The aim of this paper is to present efficient algorithms for the detection of multiple targets in noisy images of a torus. The algorithms are based on the optimal filter of a multidimensional Markov chain signal. We also present some simulations, in the case of one, two, three and four targets, showing the efficiency of the method for detecting the positions of the targets on a torus. This paper is an extension of [1].

Key words: Filtering, Markov chains, multiple targets.

Résumé

Le but de cet article est de présenter des algorithmes efficaces pour la recherche de cibles dans des images bruitées d'un tore à une dimension. Les algorithmes reposent sur le filtre optimal pour une chaîne de Markov multidimentionnelle représentant le signal. Nous présentons aussi des simulations dans le cas d'une, deux, trois et quatre cibles, démontrant l'efficacité de la méthode proposée pour la détection de positions de cibles sur le tore. Ce papier est une extension de [1].

1 Introduction

We are interested to locate m ($m \in \mathbb{N}$) targets moving in a one-dimensional torus images. Information is provided by a sequence of noisy images of the torus. As in [1], images are not crystal clear and we are using filtering theory to find a good estimation the positions of targets.

MODEL FOR TARGETS: Let $(S_k)_{k\geq 0}$ be a finite system of m targets evolving in a countable space $\mathcal{X}$. In our case we assume that $\mathcal{X}$ is a one dimensional torus, that is:

$$\mathcal{X} = \{1, \cdots, N\},$$

where we suppose that $N + 1 = 1$.

The state space of $(S_k)_{k\geq 0}$ is the set $\{0, \cdots, m\}^{\mathcal{X}}$ so that

$$\sum_{x\in\mathcal{X}} S(x) = m.$$

Targets have particular dynamics: we suppose that at each step, each target chooses the dynamics is used. There are two possibilities the first or the second dynamic, and target jump or not to an other dynamic by a Markov chain on $\{1, 2\}$. For sake of simplicity, the results stated for two, but the same method applies for any finite number of dynamics. In view to the simulations on Section 3 we consider that if a target choose the first (resp. second) dynamic it will go to the right (resp. left), with greater probability.

Let $\mathcal{A} = \mathcal{X}^m$ and $\mathcal{B} = \{1, 2\}^m$. The position of targets is exactly described by a set $A \in \mathcal{A}$. And if we note by T_k the dynamic used for the m targets at the step k, we have $T_k \in \mathcal{B}$.

For any $k \geq 0$, set

$$\xi_k = (S_k, T_k).$$

We can easily see that $(\xi_k)_{k\geq 0}$ be a finite Markov chain.

Let $S \in \{0, \cdots, m\}^{\mathcal{X}}$ such that $\sum_{x\in\mathcal{X}} \xi(x) = m$ and let $A \in \mathcal{A}$. If S is associated to the set A, that mean that each coordinate of A represent a target, then we define $F_A(S) = 1$ and if A don't represent S we set $F_A(S) = 0$. By this method we have builded a function F_A on $\{0, \cdots, m\}^{\mathcal{X}}$.

Let set now

$$M\big((A, \underline{i}), (B, \underline{j})\big) = E\Big(F_A(S_{k+1})\mathbb{1}_{\underline{i}}(T_{k+1})|F_B(S_k) = 1, \mathbb{1}_{\underline{j}}(T_k) = 1\Big), \tag{1}$$

where $A, B \in \mathcal{A}$ and $\underline{i}, \underline{j} \in \mathcal{B}$. The transition function M is important since it describes exactly the movement of the Markov chain $(\xi_k)_{k\geq 0}$.

MODEL FOR OBSERVATIONS: Observations are denoted by $Y_k \in \{0, 1\}^{\mathcal{X}}$. Given $\xi_0, \ldots, \xi_k$, we assume the $(Y_k(x))_{x\in\mathcal{X}}$ are independent and

$$P(Y_k(x) = 0|S_k(x) = 0) = p_0, \tag{2}$$

$$P(Y_k(x) = 1|S_k(x) \neq 0) = p_1, \tag{3}$$

where $0 < p_0, p_1 < 1$.

Remark. The model is not the same as [1]. In the present case, targets are moving in a one-dimensional torus and several targets can be on the same location. This is quite interesting difference because in the light of (3), observations $(Y_k)_{k\geq 0}$ don't detect if there are more than one target on a pixel.

In the next section, we will present a straightforward algorithm to compute the optimal filter. In Section 3 simulations for one, two, three and four targets, showing the performance of the algorithm proposed.

2 Description of the optimal filter

Contrary to [1], throughout this article we assume the transition function M to be known. Our aim is to find an easy algorithm to compute

$$E(F_A(S_k)|\mathcal{Y}_k),$$

where $A \in \mathcal{A}$ and $\mathcal{Y}_k$ is the sigma-algebra generated by observations $Y_1, \ldots, Y_k$. The conditional probabilities describe the estimation of the distribution of the m targets on the torus.

The first step is to compute, for any $y \in \{0,1\}^{\mathcal{X}}$, the following conditional probability

$$P(Y_k = y|\xi_k = (S,T)) = P\left(\cap_{x\in\mathcal{X}}\{Y_k(x) = y(x)\}|\, \xi_k = (S,T)\right),$$

where S is associated to a set $A \in \mathcal{A}$ and $T \in \mathcal{B}$.

Note that Y_k does't depend on the T, just on the position of targets S. Using the independence assumption, together with (2) and (3), one can check that

$$P(Y_k = y|\xi_k = (S,T)) = \prod_{x\in\mathcal{X}} \Big((1-p_0)^{y(x)} p_0^{1-y(x)}\Big)^{Q(S(x))} \Big(p_1^{y(x)}(1-p_1)^{1-y(x)}\Big)^{R(S(x))}$$

where

$$Q(x) = \prod_{i=1}^{m} \frac{x-i}{-i},$$

and

$$R(x) = \left(\frac{x}{1}\frac{x-2}{1-2}\cdots\frac{x-m}{1-m}\right) + \cdots + \left(\frac{x}{m}\frac{x-1}{m-1}\cdots\frac{x-(m-1)}{m-(m-1)}\right).$$

Polynomials $R(x)$ and $Q(x)$ satisfy

$$\begin{cases} Q(x) = 0 & \text{if} \quad x = 1, \cdots, m \\ Q(x) = 1 & \text{if} \quad x = 0 \\ R(x) = 0 & \text{if} \quad x = 0 \\ R(x) = 1 & \text{if} \quad x = 1, \cdots, m. \end{cases}$$

Thus

$$P(Y_k = y|\xi_k = (S,T)) = \left(\frac{1-p_0}{p_0}\right)^{\langle yQ(S)\rangle} \left(\frac{p_1}{1-p_1}\right)^{\langle yR(S)\rangle} p_0{}^{\langle Q(S)\rangle}(1-p_1)^{\langle R(S)\rangle},$$

where

$$\begin{cases} \langle y\rangle = \sum_{x\in\mathcal{X}} y(x) \\ \langle R(S)\rangle = \sum_{x\in\mathcal{X}} R(S(x)) \\ \langle yR(S)\rangle = \sum_{x\in\mathcal{X}} y(x)R(S(x)). \end{cases}$$

Next, set

$$\Lambda(y,\xi) == \left(\frac{1-p_0}{p_0}\right)^{\langle yQ(S)\rangle} \left(\frac{p_1}{1-p_1}\right)^{\langle yR(S)\rangle} p_0{}^{\langle Q(S)\rangle}(1-p_1)^{\langle R(S)\rangle},$$

where $\xi = (S,T)$.

Let P be the joint law of the Markovian targets with initial distribution ν, and the observations, and let Q be the joint law of the Markovian targets with initial distribution ν, and independent Bernoulli observations with mean 1/2.

Further let $\mathcal{G}_k$ be the sigma-algebra generated by $Y_1,\ldots,Y_k,\xi_0,\ldots,\xi_k$. Then it is easy to check that with respect to $\mathcal{G}_k$, P is equivalent to Q and

$$\left.\frac{dP}{dQ}\right|_{\mathcal{G}_k} = \prod_{j=1}^{k} 2^N \Lambda(Y_j,\xi_j). \tag{4}$$

Further define

$$L_k = \prod_{j=1}^{k} \Lambda(Y_j,\xi_j). \tag{5}$$

It follows that for any $\mathcal{G}_k$-measurable random variable Z and for any sigma-algebra $\mathcal{F} \subset \mathcal{G}_k$,

$$E_P\left(Z|\mathcal{F}\right) = \frac{E_Q\left(ZL_k|\mathcal{F}\right)}{E_Q\left(L_k|\mathcal{F}\right)}. \tag{6}$$

While this formula is an easy consequence of the properties of conditional expectations, in the context of filtering, (6) is known as the Kallianpur-Striebel formula.

The key observation here is to note that expectations relative to Q are much easier to evaluate since the signal and the observations are independent. Moreover all variables $\{Y_i(x)\}_{1\le i\le k, x\in I}$ are independent and identically distributed Bernoulli with mean 1/2.

For any $A \in \mathcal{A}$ and $\underline{i} \in \mathcal{B}$, define

$$q_k(A, \underline{i}) = E_Q\big(F_A(S_k)\mathbb{1}_{\underline{i}}(T_k)L_k|\mathcal{Y}_k\big).$$

Note that, according to (6), we have, for any $A \in \mathcal{A}$,

$$P\left(F_A(S_k) = 1|\,\mathcal{Y}_k\right) = \frac{\sum_{\underline{j}\in\mathcal{B}} q_k\big(A, \underline{j}\big)}{\sum_{B\in\mathcal{A},\underline{j}\in\mathcal{B}} q_k\big(B, \underline{j}\big)}.$$

Therefore the conditional law of ξ_k given $\mathcal{Y}_k$ is completely determined by the $q_k(A, \underline{i})$'s with $A \in \mathcal{A}, \underline{i} \in \mathcal{B}$.

Our goal will be attained if one can find a recursive formula for the "unnormalized measure" q_k. To this end, set

$$D_A(y) = \Lambda(y, \xi_A)$$

where $A \in \mathcal{A}$ and $\xi_A = (S_A, T)$ with S_A exactly distributed by the set $A \in \mathcal{A}$.

Using independence and identity (4), we have

$$\begin{aligned}
q_{k+1}(A, \underline{i}) &= E_Q\big(F_A(S_{k+1})\mathbb{1}_{\underline{i}}(T_{k+1})L_{k+1}|\mathcal{Y}_{k+1}\big)\\
&= E_Q\big[E\big(F_A(S_{k+1})\mathbb{1}_{\underline{i}}(T_{k+1})\Lambda(Y_{k+1}, \xi_{k+1})|Y_{k+1}, \xi_k\big)L_k|\mathcal{Y}_k\big]\\
&= \sum_{B\in\mathcal{A},\underline{j}\in\mathcal{B}} E_Q\left(E\left((F_A(S_{k+1})\mathbb{1}_{\underline{i}}(T_{k+1})\right.\right.\\
&\qquad\qquad \left.\left.\times\ F_B(S_k)\mathbb{1}_{\underline{j}}(T_k)\Lambda(Y_{k+1}, \xi_{k+1})|Y_{k+1}, \xi_k\Big)\ L_k|\mathcal{Y}_k\right)\\
q_{k+1}(A, \underline{i}) &= D_A(Y_{k+1}) \sum_{B\in\mathcal{A},\underline{j}\in\mathcal{B}} M\big((A, \underline{i}), (B, \underline{j})\big)E_Q\Big(F_B(S_k)\mathbb{1}_{\underline{j}}(T_k)|\mathcal{Y}_k\Big)\\
&= D_A(Y_{k+1}) \sum_{B\in\mathcal{A},\underline{j}\in\mathcal{B}} M\big((A, \underline{i}), (B, \underline{j})\big)q_k\big(B, \underline{j}\big)
\end{aligned}$$

Therefore we obtain the so-called "Zakai" equation

$$q_{k+1}(A, \underline{i}) = D_A(Y_{k+1}) \sum_{B\in\mathcal{A},\underline{j}\in\mathcal{B}} M\big((A, \underline{i}), (B, \underline{j})\big)q_k\big(B, \underline{j}\big). \tag{7}$$

Under this finiteness hypothesis, Zakai equation (7) is a finite sum and it can be evaluated easily (theoretically). Note that by definition, q_0 is determined by the initial law of the targets. Having observed Y_1, one can calculate the measure q_1, and so on.

The location of the m targets after the k-th observation can be estimated by choosing $A \in \mathcal{A}$, such that $Q_k(A) = \max_{C \in \mathcal{A}} \sum_{\underline{i} \in \mathcal{B}} q_k(C, \underline{i})$.

We can also estimate dynamics for targets at the k-th observation by choosing $\underline{i} \in \mathcal{B}$ such that $I_k(\underline{i}) = \max_{\underline{j} \in \mathcal{B}} \sum_{A \in \mathcal{A}} q_k(A, \underline{j})$.

Remark. In view of applications, one can either consider that the parameters p_0 and p_1 have been estimated or one can use the maximum likelihood method to estimate them from the observation of images.

3 Applications and simulations

Animations representing the results of the simulations described below, as in Figure 1 for two targets, can be obtained at the web site

http://www.lsp.ups-tlse.fr/Fp/Gentil/ensimulations2.html.

Calculations were done using C++ and MATLAB.

The targets are independent and move to the nearest neighbors, moving right or left, with probabilities α_1 and α_2 ($\alpha_1, \alpha_1 \in [0,1]$ with $\alpha_1 + \alpha_2 = 1$), if we are in the first dynamic and α_3 and α_4 if we are in the second dynamic ($\alpha_3, \alpha_4 \in [0,1]$ with $\alpha_3 + \alpha_4 = 1$).

Moreover coordinates of T_k are independent Markov chains with values in $\{1, 2\}$, with the following probability

$$\begin{cases} P(T_{k+1}(i) = 2 | T_k(i) = 1) = a_i \\ P(T_{k+1}(i) = 1 | T_k(i) = 2) = b_i, \end{cases}$$

$0 \leq a_i, b_i \leq 1$.

Due to hardware limitations, we restricted the simulations to the case of one, two, three and four targets, and torus of size 500 in the case of one and two targets, size 150 for three targets and size 25 for four targets.

In order to estimate the efficiency of our algorithm, we computed the mean error over several time intervals. We found out that the positions predictions were quite good after 5 to 10 steps. The error made at each iteration was calculated in the following way: in case of just one estimate A, we calculated the L^1-distance between the targets and the estimate; in case of several estimates, the largest L^1-distance was kept. The first iteration is never considered.

We also took into account various values of parameters p_0 and p_1, the others parameters are fixed as follow :

$$\begin{cases} \alpha_1 = \alpha_3 = 0.2 \\ \alpha_2 = \alpha_4 = 0.8 \\ a_i = b_i = 0.9. \end{cases}$$

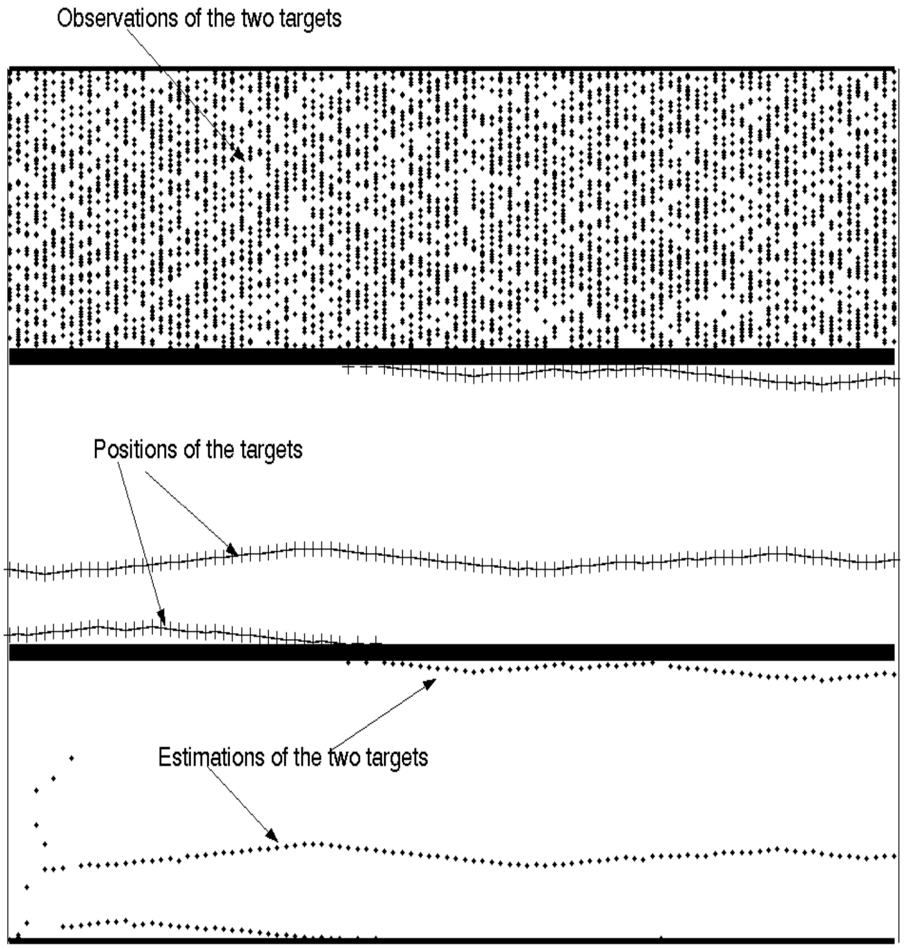

Figure 1: MATLAB film with $m = 2$, $N = 300$ and $p_0 = p_1 = 0.85$.

To finish, the initial distribution q_0 was chosen to be the uniform law on all possible configurations. The results are reported in Tables 1, 2, 3 and 4.

Table 1: *Mean error for one target in torus of size* 500.

t	$[2, 100]$	$[10, 100]$	$[30, 100]$
$p_0 = p_1 = 0.9$	3.9	0.2	0.1
$p_0 = p_1 = 0.95$	2.8	0.4	0.3

Table 2: *Mean error for two targets in torus of size* 500.

t	$[2, 100]$	$[10, 100]$	$[30, 100]$
$p_0 = p_1 = 0.9$	6.5	0.8	0.8
$p_0 = p_1 = 0.95$	5.5	0.2	0.3

Table 3: *Mean error for three targets in torus of size* 150.

t	$[2, 100]$	$[10, 100]$	$[30, 100]$
$p_0 = p_1 = 0.9$	2.2	1.8	1.3
$p_0 = p_1 = 0.95$	1.0	0.4	0.4

Table 4: *Mean error for four targets in torus of size 25.*

t	$[2, 100]$	$[10, 100]$	$[30, 100]$
$p_0 = p_1 = 0.9$	5.1	3.0	3.2
$p_0 = p_1 = 0.95$	0.6	0.5	0.6

Note that, in each case, the algorithm shows really satisfactory results. From the 5-th or 10-th iteration (depending on the choice of parameters p_0 and p_1), the distance between the estimation and the targets is about one or three pixels. This is due to the fact that the algorithm provides an exact solution to the resolution of the optimal filter. There is a little loss when we have a lot of targets, this is due to the fact that we use the L^1 distance to calculate the error.

Acknowledgments

Partial funding in support of this work was provided by the Natural Sciences and Engineering Research Council of Canada, by the MITACS-PINTS Network of Center of Excellence, Lockheed Martin Canada and by the Institut de finance mathématique de Montréal.

References

[1] Gentil, I., Rémillard, B. and Del Moral, P. (2002). Filtering of images for detecting multiple targets trajectories, preprint.