Point Estimation for Jump Markov Systems: Various MAP Estimators

Yvo Boers and Hans Driessen
Thales Nederland B.V.
Hengelo, The Netherlands

Arun Bagchi
Department of Applied Mathematics
University of Twente
Enschede, The Netherlands

Abstract – In this paper we will provide methods to calculate different types of Maximum A Posteriori (MAP) estimators for jump Markov systems. The MAP estimators that will be provided are calculated on the basis of a running Particle Filter (PF). Furthermore, we will provide convergence results for these approximate, or particle based estimators. We will show that the approximate estimators convergence in distribution to the true MAP values of the stochastic variables. Additionally, we will provide an example based on tracking closely spaced objects in a binary sensor network to illustrate some of the results and show their applicability.

Keywords: Particle filters, Maximum a Posteriori Estimators, Dynamical Systems, Target Tracking

1 Introduction

The focus of this paper is point estimators for stochastic dynamical systems. The application area of this topic is naturally very wide. The scope covers all sorts of fields, such as target tracking, navigation, communication, image processing or finance, just to name a few. In fact, any problem that can be formulated as a dynamical system on which measurements are performed falls in the application area. The process of inferring information on the state of such a system based on measurements is called filtering.

Point estimators are commonly used. Examples of popular point estimators are the minimum variance (MV) estimator, often also called Minimum Mean Squared Error (MMSE) estimate or the maximum a posteriori (MAP) estimator. It is well known that in case of linear and Gaussian systems, for which the a posteriori densities are Gaussian, the two coincide. However, for general, non-linear and/or non-Gaussian systems they generally do not coincide. An illustration of this is provided in figure 1.

In modern Bayesian estimation theory and more specifically in particle filtering, see e.g. [1], one might argue or question: why concentrate on the topic of point estimation at all? Especially when the entire a posteriori density is provided anyway, e.g. by means of the particles of a running particle filter. Because it is well known that this a posteriori density is the maximum of information that can be obtained. The answer is very simple: because a lot of practical and operational applications require a point estimate. Moreover, it might be far from trivial to give an interpretation of the entire a posteriori density or the particle cloud. For example, in the case of target tracking, when the filter describes one single target, an interpretation might be not that difficult. However, when the filter describes a multi-target setting, possibly including a description of the number of targets as well, it is far from trivial to provide a good interpretation. Having said all this, we do emphasize that in the scope of this paper the entire underlying...
description, i.e. the a posteriori probability density, is maintained by a running particle filter. Thus, the MAP estimates provided in this paper are for output reasons only and do not comprise the running particle filter in any way whatsoever.

As mentioned before, in certain applications a point estimator such as the MAP estimator can be very useful. One such application is e.g. tracking of closely spaced objects, see [2] and [3]. Consequently, in [2] it has been shown quantitatively that for the application at hand the MAP estimator is superior to the MV estimator. In general, we can say that for non-linear and/or non-Gaussian systems MAP estimation is a useful alternative for MV estimation. Some other applications, in which multi-modality naturally arises, are terrain aided navigation, see e.g. [4], and tracking in binary sensor networks, see e.g. [5]. However, there are undoubtedly many more applications that will lead to multi-modality and for which the MAP estimator will be a good alternative for the MV estimator.

The particle filter based MAP estimator for non-linear dynamical systems has first been presented in [6]. In [6] and [7] it has been shown that based on a running particle filter the MAP estimator can be calculated. Moreover, explicit formulas and an algorithm for calculating this PF based MAP estimator have been provided. Previous to [6] and [7] the particle filter based MAP estimator has been erroneously thought to be equal to the particle with the highest weight. This idea has been proven wrong in [7] and correct expressions and algorithms to calculate the true MAP estimator have been given therein.

Further, it has been proven in [7] that the particle based MAP estimator converges to the true MAP estimator. It is good to note that such a proof does not exist for the regularly used MV estimator, see also [8]. Recently, for the MV estimator convergence has been proven. However, at the cost of additional assumptions on the problem at hand.

Also, the earlier work [9] should not be left unmentioned. In this paper a particle based MAP estimator for the whole sequence or trajectory has been presented. Thus, the maximum of $p(s_0, s_1, \ldots, s_k \mid Z_k)$ is approximated. The result of this procedure is a maximizing sequence: $\hat{s}_{0:k}$. This sequence MAP estimator is obtained by employing a Viterbi optimization at every time step. The complexity of the problem in [9] naturally grows with the time horizon $k$. In this paper the focus is to obtain a MAP estimate for the current state only. As a result, we are interested in the MAP estimate associated with the filtering density $p(s_k \mid Z_k)$. We also stress the fact that the MAP estimate of the filtering density is not necessarily the same as taking the $k^{th}$ element of the sequence MAP estimator. Especially in the case of multi-modal a posteriori distributions, i.e. a case of particular interest in MAP estimation, the estimates do not have to coincide at all. See [10] for an actual example of this and some more explanation and discussion.

As mentioned before, the scope of this paper is MAP estimation for jump Markov dynamical systems.

The contributions of this paper are:

- The formulation of an array of relevant MAP estimators for a wide class of jump Markov nonlinear dynamical systems
- Analytical closed form formulas for these MAP estimators.
- Particle filter based approximations for these MAP estimators.
- Convergence results for these approximations.

In a sense the results presented in this paper can be seen as a natural extension or generalization of the results presented in [6] and [7]. The extension lies primarily the consideration of a more general class of systems, i.e. class of non-linear hybrid jump Markov systems.

## 2 System setup

Let us consider the following jump Markov nonlinear discrete time dynamical system

$$s_k = f(m_k, s_{k-1}, w_{k-1}), \ k \in \mathbb{N}$$  \hspace{1cm} (1)

$$\{m_k\}_{k \in \mathbb{N}}$$ is Markov with transition matrix $\Pi_k$  \hspace{1cm} (2)

$$z_k = h(s_k, m_k, v_k), \ k \in \mathbb{N}$$  \hspace{1cm} (3)

with the initial state being distributed according to $p_0(m_0, s_0)$ where

- $s_k \in \mathcal{S} \subset \mathbb{R}^n$ is the continuous state of the system.
- $m_k \in \mathcal{M} \subset \mathbb{N}^m$ is the discrete state of the system.
- $z_k \in \mathcal{R}^p$ is the measurement.
- $w_k$ is the process noise and $p_{w_k}(w)$ is the probability distribution of the process noise.
- $v_k$ is the measurement noise and $p_{v_k}(v)$ is the probability distribution of the measurement noise.

Furthermore, we define by $Z_k = \{z_0, \ldots, z_k\}$, the measurement history.

The filtering problem amounts to calculating or reconstructing the a posteriori density, also called the filtering density. Hence, in case of a jump Markov system, this would be the reconstruction of $p(s_k, m_k \mid Z_k)$. 

34
3 Multiple Model MAP Estimators

Associated with the previously introduced filtering solution, which amounts to the reconstruction of an entire probability density function, we will now define several useful MAP point estimators.

- MAP for the joint base state and modal state: JMAP
  \[ p(s_k, m_k | Z_k) \] (4)

- MAP for the marginalized base state: MBMAP
  \[ p(s_k | Z_k) \] (5)

- MAP for the marginalized modal state: MMMAP
  \[ p(m_k | Z_k) \] (6)

- MAP for the marginalized modal state conditioned base state: MCBMAP
  \[ p(s_k | m_k, Z_k) \] (7)

- MAP for the marginalized base state conditioned modal state: MBCMMAP
  \[ p(m_k | s_k, Z_k) \] (8)

We will now proceed with the derivation of particle based approximations of the different types of MAP estimators, that we have just defined.

The standing assumption is that we have a running multiple model particle filter for our Jump Markov system. In section 3.5.5. of [1] a detailed description of such an algorithm has been provided.

Such a filter can generally be characterized by the set of triplets:

\[ \{s^i_k, m^i_k, q^i_k \}_{i=1...N} \] (9)

where \( s^i_k \) is the base state part of particle \( i \) at time step \( k \), \( m^i_k \) is the modal state part and \( q^i_k \) the corresponding weight for this particle. Furthermore, it is assumed that the weights \( q^i_k \) have been normalized and sum up to one.

We will now derive the different MAP estimators introduced above. Moreover, these derivations lead to approximation schemes. Through these schemes the different MAP estimators can be reconstructed approximately, based on a running multiple model particle filter, characterized by (9).

3.1 Derivation of the JMAP

We are interested in the MAP estimator associated with the probability density function \( p(s_k, m_k | Z_k) \). We can write:

\[ p(s_k, m_k | Z_k) \propto p(z_k | s_k, m_k)p(s_k, m_k | Z_{k-1}) \] (10)

we can directly evaluate \( p(z_k | s_k, m_k) \), thus we concentrate on \( p(s_k, m_k | Z_{k-1}). \)

\[ p(s_k, m_k | Z_{k-1}) = \int_{S \times M} p(s_k, m_k | s_{k-1}, m_{k-1}) \] (11)

\[ p(s_{k-1}, m_{k-1} | Z_{k-1}) ds_{k-1} dm_{k-1} \]

where \( S \) denotes the state space for the base state and \( M \) the state space for the modal state. However, we note that the modal state can only assume discrete values, therefore the integral over \( M \) in (11) reduces to a sum and we arrive at

\[ p(s_k, m_k | Z_{k-1}) = \sum_{m_{k-1} \in M} \int_{S} p(s_k, m_k | s_{k-1}, m_{k-1}) \] (12)

\[ p(s_{k-1}, m_{k-1} | Z_{k-1}) ds_{k-1} \]

and by using

\[ p(s_k, m_k | s_{k-1}, m_{k-1}) = p(s_k | s_{k-1}, m_{k-1}, m_k)p(m_k | s_{k-1}, m_{k-1}) \]

we obtain

\[ p(s_k, m_k | Z_{k-1}) = \sum_{m_{k-1} \in M} \int_{S} p(s_k | s_{k-1}, m_{k-1}, m_k) \] (13)

\[ p(m_k | s_{k-1}, m_{k-1})p(s_{k-1}, m_{k-1} | Z_{k-1}) ds_{k-1} \]

which in turn is the same as

\[ p(s_k, m_k | Z_{k-1}) = \sum_{m_{k-1} \in M} \int_{S} p(s_k | s_{k-1}, m_k) \] (14)

\[ p(m_k | m_{k-1})p(s_{k-1}, m_{k-1} | Z_{k-1}) ds_{k-1} \]

Now, given the fact that we have a particle representation for \( p(s_{k-1}, m_{k-1} | Z_{k-1}) \) through \( \{s^i_{k-1}, m^i_{k-1}, q^i_{k-1}\}_{i=1...N} \), we can approximate (14) by

\[ p(s_k, m_k | Z_{k-1}) \approx \sum_{i=1}^{N} p(s_k | s^i_{k-1}, m_k)p(m_k | m^i_{k-1})q^i_{k-1} \] (15)
The above quantity can be evaluated in the sense that for every pair \((s_k, m_k)\), based on the particle set at time \(k - 1\) and the system predictive densities, both for the base and the modal state, it can be numerically evaluated. Thus, going back and starting with the approximate equality (15), we can approximate \(p(s_k, m_k \mid Z_k)\) in the same manner as \(p(s_k \mid Z_k)\) has been approximated in [7].

### 3.2 Derivation of the MBMAP

To derive the MBMAP, we start out with the JMAP result

\[
p(s_k, m_k \mid Z_{k-1}) \approx \sum_{i=1}^{N} p(s_k \mid s^i_{k-1}, m_k)p(m_k \mid m^i_{k-1})q^i_{k-1}
\]

now integrating both sides of the equation w.r.t to the mode variable leads to:

\[
\sum_{m_k \in M} p(s_k, m_k \mid Z_{k-1}) \approx \sum_{m_k \in M} \sum_{i=1}^{N} p(s_k \mid s^i_{k-1}, m_k)p(m_k \mid m^i_{k-1})q^i_{k-1}
\]

Changing the order of summation in (17) provides the result.

### 3.3 Derivation of the MMMAP

We are interested in the MAP estimator associated with the probability density function \(p(m_k \mid Z_k)\).

We can write:

\[
p(m_k = m \mid Z_k) = \int_{S} p(s_k, m_k = m \mid Z_k)ds_k
\]

which is readily approximated by:

\[
\int_{S} p(s_k, m_k = m \mid Z_k)ds_k \approx \sum_{i : m_i = m} q_k^i
\]

Thus, this quantity is obtained easily based on the particle weights at time \(k\).

### 3.4 Derivation of the MMCBMAP

We are interested in the MAP estimator associated with the probability density function \(p(s_k \mid Z_k, m_k)\).

This MAP estimator is readily approximated by using the identity:

\[
p(s_k \mid Z_k, m_k) = \frac{p(s_k, m_k \mid Z_k)}{p(m_k \mid Z_k)}
\]

and exploiting the previously derived approximative relations for the JMAP and the MMMAP.

### 3.5 Derivation of the MBCMMAP

We are interested in the MAP estimator associated with the probability density function \(p(m_k \mid Z_k, s_k)\).

This MAP estimator is readily approximated by using the identity:

\[
p(m_k \mid Z_k, s_k) = \frac{p(s_k, m_k \mid Z_k)}{p(s_k \mid Z_k)}
\]

and exploiting the previously derived approximative relations for the JMAP and the MBMAP.

### 4 Convergence of the estimators

Having derived approximation schemes for the different types of MAP estimators, a natural question to ask is: how do these approximations relate to the true MAP estimators? Moreover, what is the behavior or quality of the approximation as the number of particles grows and ultimately grows to infinity. Hopefully the approximations of the MAP estimators tend towards the true MAP estimators for the number of particles going to infinity. In order to answer these questions, we will investigate convergence properties of the approximate MAP estimators.

Also in this section, we will concentrate on convergence properties of the different approximate MAP estimators. The type of convergence that we consider (we call it weak convergence) will be defined below and is the exact same one that has been used in [8].

For the single model case convergence results for the MAP estimator have been provided in [7].

**Definition 4.1**

We define by \(C_b(D)\) the set of all continuous bounded functions on \(D\). Thus, \(f : D \to \mathbb{R} \in C_b(D)\) if and only if:

- \(f\) is continuous
- There exists a number \(R \in \mathbb{R}^+\) such that for all \(x \in D\) \(|f(x)| \leq R\)

**Remark 4.2**

As it only makes sense to define continuity on the non-discrete part of the state space, definition 4.1, must be read with respect to that part of the state space only. Thus, in case of \(D = S \times M\), the functions in the set \(C_b(D)\) are bounded on \(S \times M\) and continuous on \(S\).

For example, in case of the JMAP the set \(D\) would be defined as \(D := S \times M\) and in case of MBMAP we would have to set \(D := S\).

We will now define convergence. We do this with the JMAP in mind, thus for the case of \(D = S \times M\), but it can be defined for all other instances as well.
Definition 4.3 (Weak Convergence)
Suppose we have a running particle filter, characterized by \( \{ s^i_k, m^i_k, q^i_k \}_{i=1}^{N} \), see also (9). Then we say that the particle filter converges to the true, but unknown a posteriori distribution of interest \( p(s_k, m_k | Z_{k-1}) \) if and only if for all \( g(s, m) \in D \):

\[
\lim_{N \to \infty} \sum_{i=1}^{N} g(s^i_k, m^i_k)q^i_k = \int_D g(s_k, m_k)p(s_k, m_k | Z_k)ds_kdm_k
\] (22)

This type of convergence is the weakest type of stochastic convergence. It is sometimes also referred to as convergence in distribution or convergence in law. See e.g. [11] for a good overview of different types of stochastic convergence and their relationships.

In [8], this type of weak convergence has been proven to hold for particle filters. It has also been reported that for the most well known case of a point estimator, i.e. the MV estimator, convergence cannot be proven along this line. The reason for this is that for this proof the function \( g \) would have to be the identity, which is not bounded. Recently in [12], under additional assumptions, proofs of convergence for unbounded functions and thus also for the MV estimator have been provided.

Below we will show convergence of different MAP estimators based on the bounded function assumption. Thus, relying on the results from [8].

We will be able to prove convergence and the key element we need for that is that we will be able to prove that the particle approximations for \( p(\cdot | Z_{k-1}) \) in the different MAP derivations actually converge to \( p(\cdot | Z_{k-1}) \). The remainder is less interesting and quite straightforward. The rationale behind this is that once convergence of the particle approximation of \( p(\cdot | Z_{k-1}) \) has been proven, also the maximizers and the maximizing arguments of the particle approximations converge to the true maximizers and maximizing arguments. We refer also to [7] for a more formal proof of these statements.

Next, we will now concentrate on the convergence of the particle approximations for \( p(\cdot | Z_{k-1}) \). We will do this for the JMAP and the MBMAP. The others are either less involved or can be easily derived based on the JMAP and MBMAP.

Theorem 4.4 (Convergence of the JMAP)
Assume that we have a running particle filter for a jump Markov system. The particle filter is characterized by the triplet \( \{ s^i_k, m^i_k, q^i_k \}_{i=1}^{N} \). Furthermore, assume that all transition probability functions are bounded. The following now holds:

\[
\lim_{N \to \infty} \sum_{i=1}^{N} p(s_k | s^i_{k-1}, m_k)p(m_k | m^i_{k-1})q^i_{k-1} = p(s_k | Z_{k-1})
\] (23)

Proof

\[
\lim_{N \to \infty} \sum_{i=1}^{N} p(s_k | s^i_{k-1}, m_k)p(m_k | m^i_{k-1})q^i_{k-1} = \lim_{N \to \infty} \sum_{i=1}^{N} \Psi(s_k, m_k, s^i_{k-1}, m^i_{k-1})q^i_{k-1} = \int_{S \times S \times M} p(s_k | s_{k-1}, m_k)p(m_k | m_{k-1}) \sum_{m \in M} p(s_{k-1}, m_{k-1} | Z_{k-1}) ds_{k-1}dm_{k-1}
\]

which is equal to \( p(s_k, m_k | Z_{k-1}) \). This last step is easily verified by working back through the equations starting from equation (14).

\[
\text{Theorem 4.5 (Convergence of the MBMAP)}
\]
Assume that we have a running particle filter for a jump Markov system. The particle filter is characterized by the triplet \( \{ s^i_k, m^i_k, q^i_k \}_{i=1}^{N} \). Furthermore, assume that all transition probability functions are bounded. The following now holds:

\[
\lim_{N \to \infty} \sum_{i=1}^{N} \sum_{m \in M} p(s_k | s^i_{k-1}, m_k)p(m_k | m^i_{k-1})q^i_{k-1} = p(s_k | Z_{k-1})
\] (26)

For the sake brevity we refer the interested reader to [10] for the proof of theorem 4.5.

Convergence for the MMCBMAP and the MBCCMAP is inherited from JMAP and MBMAP convergence.

Furthermore, we do stress the fact that in most applications the assumption that the transition probability densities are bounded is satisfied. One field of particular interest is e.g. the field of target tracking. In most of these applications the transition probability densities are assumed to be Gaussian and thus, naturally bounded. See e.g. [1]. However, in general, it is very well possible to construct densities that are not bounded.
5 Simulations

In this simulation example we have used the binary sensor network system that has been presented in [5].

For the sake of completeness, we will provide the sensor model here once more.

For the measurement part of the system: for a single sensor $j \in \{1, \ldots, M\}$, where $M$ is the number of sensors, the measurement is binary, i.e. either 0 or 1 and the likelihood is, dropping the subscript for the time step:

$$p(z_j|s) = 1_{R_j}(s)[z_j^d(p_d^j + (1-p_d^j)p_{fa}^j)] + (1-z_j^d)(1-p_d^j)(1-p_{fa}^j) + (1-1_{R_j}(s))[z_j^d p_{fa}^j + (1-z_j^d)(1-p_{fa}^j)]$$

where $1_{R_j}(s)$ is the indicator function, assuming either a value of 1 or 0, indicating whether the state of the object is inside the sensing range of sensor $j$ or not. Furthermore, $p_d^j$ and $p_{fa}^j$ are the detection and false alarm probabilities of sensor $j$.

The total likelihood $p(z|s)$ is obtained as the product of the individual likelihoods.

Furthermore, we model a three mode system. The first mode of the system corresponds to the situation that there is no object present, the second one, that there is one object present and the third one that two objects are present. The system model allows transition from one mode to another, according to a Markov process, just as explained in section 2. For the example the Markov transition probabilities of staying in the same mode are 0.9 and the probabilities of going to another mode are 0.05.

The false alarm probability for each sensor is 0.05 and the probability of detection is 0.9. In the set up 144 sensors have been used. The sensing range has been set to 450m and the sensor are uniformly spaced over the coverage area, the sensors are placed 500m apart.

In the example 10,000 particles have been used. We emphasize here, that probably one could use less particles and maybe far less with an efficient or optimized particle filter, although even with 10,000 particles the filter runs still quite fast. However, the goal here is not numerical efficiency, but merely the illustration of the multiple model MAP estimation. Ergo, we use a plain vanilla multiple model particle filter as e.g. described in section 3.5.5. of [1]. In addition, we want to be sure to eliminate effects of a (too) low number of particles from the simulations all together.

In the figures below we show the trajectories of two objects. The objects are present throughout the course of the simulation. The black up/downward pointing triangles are the MV output, whereas the blue left/right pointing triangles are MAP outputs. Furthermore, the particle cloud(s) are shown as well in the figures. In this example the MAP estimator show for the position is the MMCBMAP, thus the the MAP estimate conditioned on a mode. The actual mode it has been conditioned on, is in turn determined by the MMMAP, see figure 7.

It can be seen that at time steps 4 and 42 the MMCBMAP states that the one target mode is the most probable. For the other time steps shown, the two target mode is most probable, at least according to the MMCBMAP.

Furthermore, it is nicely illustrated here that the MMCBMAP estimator provides much better state estimates for the position than the MV estimator. The estimates obtained by the MV estimator are not even close to the true object positions. The MV estimates are so far off in this example that there is not even a need for making a quantitative comparison between the MAP and the MV output. The deteriorated performance of the MV estimate is due to the so called mixed labelling problem. This problem has been extensively studied in [3] and indeed MAP estimation instead of MV estimation is a good way of dealing with this problem. The obvious cure when employing the previously derived approximate MAP estimates is that the approximate MAP estimate is represented by one single particle and thus inherently mixed labelling has no influence.

6 Conclusions

In this paper, we have derived several PF based MAP estimators for jump Markov dynamical systems. It has
been shown that based on a running particle filter approximate MAP estimators can be readily calculated. In addition it has also been shown that these approximate MAP estimators converge to the true MAP values of the densities at hand in a weak sense for the number of particles going to infinity. Furthermore, the value of MAP estimation has been illustrated by means of an illustrative tracking example. In the example we considered the case of tracking multiple closely spaced objects.
7 Acknowledgements

This research has been partially financially supported by the Netherlands Organisation for Scientific Research (NWO) under the Casimir program, contract number 018.003.004. Under this grant, the first author also holds a part-time position at the Stochastic Systems and Signals group of the Applied Mathematics Department at the University of Twente.

The authors thank Dr. Ir. Martin Podt of Thales Nederland B.V. for the careful reading of the document and the helpful comments.

The authors also wish to thank Ms. Mary Ann Bjorklund for careful proof reading of the document and for correcting grammatical and spelling errors. Any errors left are the sole responsibility of the authors.

References


