Proceedings of the 2010 Winter Simulation Conference
B. Johansson, S. Jain, J. Montoya-Torres, J. Hugan, and E. Yücesan, eds.

ROBUST ESTIMATION OF MULTIVARIATE JUMP-DIFFUSION PROCESSES VIA DYNAMIC PROGRAMMING

Andrey Torzhkov
Puneet Sharma
Amit Chakraborty

Siemens Corporate Research
755 College Road East
Princeton, NJ, 08540, USA

ABSTRACT

In this work we present a framework for estimation of a rather general class of multivariate jump-diffusion processes. We assume that a continuous unobservable linear diffusion processes system is additively mixed together with a discrete jump processes vector and a conventional multi-variate white-noise process. This sum is observed over time as a multi-variate jump-diffusion time-series. Our objective is to identify realizations of all components of the mix in a robust and scalable way. First, we formulate this model as an Mixed-Integer-Programming (MIP) optimization problem extending traditional least-squares estimation framework to include discrete jump processes. Then we propose a Dynamic Programming (DP) approximate algorithm that is reasonably fast & accurate and scales polynomially with time horizon. Finally, we provide numerical test cases illustrating the algorithm performance and robustness.

1 INTRODUCTION

Robust estimation of linear diffusion systems is a well-studied topic which have found numerous applications in various industries. From classical least-squares problems to modern large-scale constrained l_2-norm optimization, the common solution method is to employ convex Quadratic Programming (QP) techniques. QP algorithms are accurate & scalable but are rather restrictive to extend beyond linear continuous systems. Particularly, introducing discrete events into estimation problem contradicts convexity in all but rare cases.

Discrete events are quite common in many industrial applications. For example, discrete controls (e.g. switching capacitor banks) are often used in electrical grids and need to be included into real-time grid state estimation, the corner-stone of all further network analysis and control applications. Next, discrete-space Markov chains estimation is rather common in healthcare disease progression models and manufacturing jobshop workflows. In this particular work, we draw our motivation from another important industrial engineering application, namely, a model-based failure detection based on a real-time sensor data.

It is typical for many sensor systems time-domain data to contain short-term discrete perturbations (e.g. due to shutdowns of malfunctioning) and/or medium-term regime switching (e.g. due to regular system adjustments). Also, such events often come undocumented so that it is not possible to mark them with certainty based on external knowledge. Moreover, discrete events may not just obscure sensor response to a hidden failure but themselves may carry important information about failure type and presence. The Figure 1 represents a rather typical example supporting both of these statements: it shows a slow progress of a fan failure in an air ventilation system in a coal mine using multiple spatially-distributed sensors measuring methane concentration over time. Observations clearly show presence of discrete perturbations completely obscuring any systemic trends whatsoever. As a subsequent offline engineering analysis confirmed, in this example both the slow growing systemic trend and the discrete perturbations contained a failure signature that was nearly impossible to find in their mix observed through the sensors.

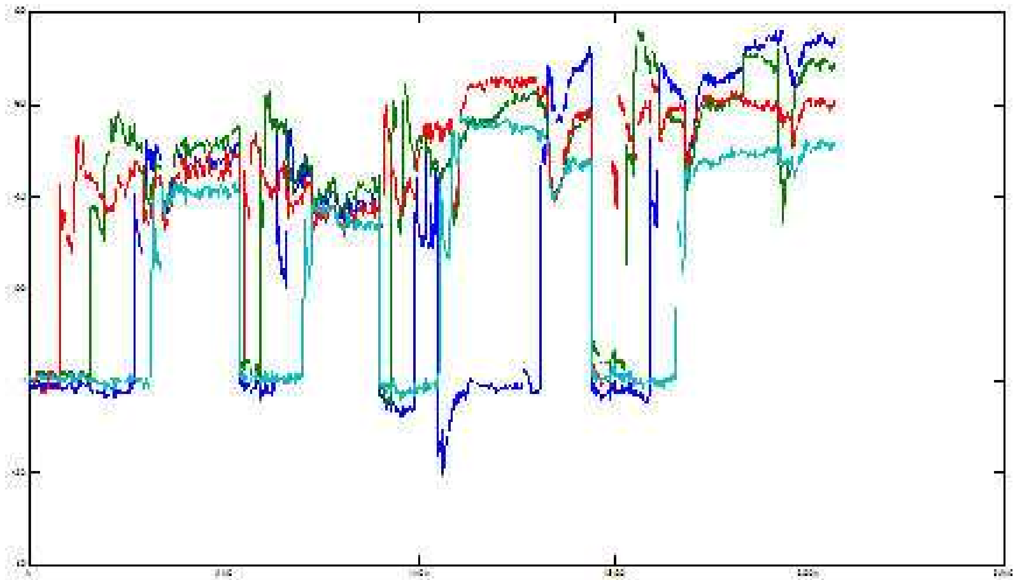

Figure 1: An example of a failure progress disguised by large perturbations.

The principal difference in statistical modeling of discrete/jump processes as compared to continuous processes is to include jumps count, durations and magnitudes as explicit model variables. The statistical identification approach that we take in this work is exactly the following: incorporate these variables into both the model's objective function (e.g. as max-likelihood terms working as a penalty for jumps frequency and/or durations) and into the identification constraints (i.e. domain-knowledge relations limiting jump process' magnitudes at all time points within a particular jump duration to be equal).

To illustrate our approach, consider a classic Poisson process as an example of a pure-jump process: the Poisson count process $N(t)$ is observed over a fixed time horizon T. The count process $N(t)$ is often considered in terms of its sojourn time durations t_k conditioned upon a Poisson process' count K accumulated over time T. This count's marginal distribution(for any T) is Poisson and the joint distribution of sojourn times conditioned on this count is i.i.d. uniform (for the proof see Wolff (1989)). Then, the log-likelihood function is separable/additive and contains: a non-linear probabilistic penalty for the count K and linear penalties for the (log-scaled) durations t_k. However, these variables are not directly observable but rather must be identified from regularly-observed realizations of $N(t)$. Using a well-known proxy $I(N(t+\delta t)-N(t)>0)$ instead of t_k is problematic both from theory (indicator is not a continuous function) and practical (any regular white-noise invalidates this indicator). Instead, we use regular measurement time-series $N(t_n)$ and introduce identification constraints $N(t_n)-k=0, t_{k-1}\leq t_n\leq t_k$.

The simple formulation in the Poisson process example is yet not a well-defined statistical optimization problem and our work presents a way to further reformulate & generalize it into an instance of the Mixed-Integer-Programming (MIP) models and propose a Dynamic-Programming (DP) approximate algorithm that can be used for real-time identification and scales very well with the problem size (in particular, time horizon T). Our DP algorithm is recursive backward-over-time (typical for dynamic programming) and solves only well-defined convex problems at each iteration of the recursion. We demonstrate on a simulated numerical case (combining both continuous trend, discrete perturbations & white-noise) the quality of this approximation as compared to a typical benchmark QP-based identification method that completely ignores discrete perturbations.

Benchmarking against a conventional QP/least-squares methods pursues to show the robustness of our approach. We understand robustness in its conventional meaning: identification should not vary much for a variety of independently taken measurement collections drawn from the same statistical model case. Traditionally, one of the best ways to improve robustness is to include constraints representing domain knowledge information about the nature of the data. For example, Samar, Torzhkov, and Chakraborty (2008) incorporated knowledge of a monotonic nature of a hidden continuous failure process F_t by assuming that F_t follows a random-walk with exponentially distributed increments. This assumption produced an inequality constraint $F_{t+1} \geq F_t$ and a linear penalty $\lambda \cdot (F_{t+1} - F_t)$ in the identification model while still keeping it within QP-class. In turn, the estimated sample path realization of $F(t)$ were completely insensitive to large short-term noise spikes (e.g. due to a high volatility cluster) and the identification model was robust in this sense. However, it was still possible to confuse this model with a small series of significantly long deterministic perturbations in a historic estimation mode or even a single perturbation in a moving-horizon mode.

Cleaning the data off perturbations is mostly not an option in the modern autonomous & automated systems world and is dangerous in general as it may wipe off the tracks of a failure. Including discrete perturbations A_t on top of F_t into the identification model should remove sensitivity to such outliers data but increases statistical model's degrees of freedom so it may actually distort the estimation of the core failure process $F(t)$. Thus, we note that we consider a jump-diffusion statistical estimation/identification procedure robust only if estimates of A_t indeed capture the true perturbation/jump sample path realization levels and produce the accurate track of the failure process F_t.

In this paper, we start from the same model as given in Samar, Torzhkov, and Chakraborty (2008) and we gradually add more features into that such that it eventually leads to a complete well-defined MIP-model formulation to identify at every point of time all components from an observable sum of a continuous multi-variate failure process $F(t)$, discrete perturbation process $A(t)$ and white-noise $\varepsilon(t)$.

2 LITERATURE REVIEW

It is nearly impossible to provide a comprehensive review of the stochastic processes & systems estimation research stream. Therefore, we only include a few particularly relevant references highlight distinguishing features of our model and refer to alternative approaches to the estimation problem class involved.

A large stream of research works on sample-path estimation of jump-diffusion processes comes from quantitative finance field. Lee and Mykland (2008) follows a non-parametric jump detection approach similar to ours and propose a statistical test for a class of jump-diffusion processes where jumps times are following a Possion process but jump sizes are normally distributed. Jiang and Owen (2007) proposes a Generalized Method of Moments estimator for a similar model.

Failure detection field also influences & pushes further the jump-diffusion estimation research frontier. For a review, see Gertler (1988) survey or a more recent one Venkatasubramanian et al. (2003). Seminal paper Dufour and Bertrand (1994) extended Kalman filtering framework for Markov-modulated linear systems with partially observable jumps which was recently improved by Mahmoud and Shi (2004). (Plageman, Fox, and Burgard 2007) proposes a particle filter estimator for a hybrid jump-continuous failure detection problem. We note that the area jump-continuous failure detection is still relatively scarcely explored.

Generic estimation methodology papers mostly focus on improving time-series analysis by including jump processes either into the series trend or as a regime modulator. Zhao and Wei (2003) studies a particular time series model with jump-modulated trend and proposes a weighted estimator similar to Gaussian kernel methods.

More broad non-parametric estimation view comes from machine learning field proposing non-parametric estimates either using kernel estimation or transition probability matrix extraction. For instance, Blackmore et al. (2008) develop an active method for discrimination between discrete control modes of jump Markov linear systems and use it to learn the whole structure of the estimated system. Bandi and Nguyen (2003) propose non-parametric functional estimation of infinitesimal conditional moments of jump-diffusion systems.

3 ESTIMATION MODEL

We start with a diffusion-only model that was developed in Samar, Torzhkov, and Chakraborty (2008). It is rather representative of constrained QP estimation models and provides some intutition as well as a test base.

As given in Samar, Torzhkov, and Chakraborty (2008), we estimate a realized sample path of a hidden failure process model $\{F_t\}_{t=1}^{T}$ that is known to be monotonically growing. We assume a given observations vector time series $\{Y_{i,t}\}_{i=1,t=1}^{N,T}$ that is linearly sensitive to the failure process

$F_t : Y_{i,t} = S_i \cdot F_t + \varepsilon_{i,t}$, where failure signature vector S_i measures relative strength of the sensors response and the noise vector $\varepsilon_{i,t}$ is assumed to be i.i.d. (over time t) Gaussian. As we mentioned earlier, monotonicity is modeled by assuming that the increments $F_{t+1} - F_t$ are i.i.d. and exponentially distributed with intensity λ. Given that, the realized sample path of F_t can be filtered out of the sensor data $Y_{i,t}$ via a maximum posterior likelihood estimate based on the following constrained QP model:

$$\min_{F_t} L(F|Y) = \sum_{t=1}^{T} \sum_{i,j=1}^{N} (Y_{i,t} - S_i \cdot F_t) \cdot Q_{ij}^{-1} \cdot \left(Y_{j,t} - S_j \cdot F_t\right) + \lambda \cdot (F_T - F_1) \qquad (1)$$
$$s.t. \quad F_{t+1} \geq F_t$$

where Q is the sensor noise covariance matrix.

The extension that we propose introduces a vector time series $\{A_{i,t}\}_{i=1,t=1}^{N,T}$ of pure-jump perturbation processes such that the sensors are sensitive to $F_t + A_{i,t}$rather than F_t. Key difference between F_t and $A_{i,t}$ is that: i) the log-likelihood function contains penalties for both perturbation levels $\{A_{i,k}\}_{i=1,k=1}^{N,K_i}$, durations $\{\overline{T}_{i,k} - \underline{T}_{i,k}\}_{i=1,k=1}^{N,K_i}$ and number $\{K_i\}_{i=1}^{N}$ of perturbations and ii) the constraint set includes flat level constraint $A_{i,t} = \sum_{k=1}^{K_i} A_{i,k} \cdot I\left[\overline{T}_{i,k} \leq t < \underline{T}_{i,k}\right]$ where $\overline{T}_{i,k}$ and $\underline{T}_{i,k}$ are start and end time for the k-th perturbation for the i-th sensor data.

We define the stacked vector $X_{i,k} = (A_{i,k-1}, A_{i,k}, \overline{T}_{i,k} - \underline{T}_{i,k}, 1)$ and use the following quadratic form of the modified log-likelihood objective function:

$$\min_{F_t, X_k} \sum_{t=1}^{T} \sum_{i,j=1}^{N} (Y_{i,t} - S \cdot F_t - A_{i,t}) \cdot Q_{ij}^{-1} \cdot \left(Y_{j,t} - S_j \cdot F_t - A_{j,t}\right) + \lambda \cdot (F_T - F_1) + \sum_{i=1}^{N} \sum_{k=1}^{K_i} X_{i,k}^T \cdot D \cdot X_{i,k} + \sum_{i=1}^{N} c(K_i) \cdot K_i \qquad (2)$$

where matrix D is a way to capture the perturbation transition rates specifics and $c(K_i)$captures the jump count process distribution.

We note, that the matrix D need not be semi-definite positive and the costs $c(K_i)$ need not be constant for our DP approximation to work. We also note that some of the sums, for example, $\sum_{k=1}^{K_i} X_{i,k}^T \cdot D \cdot X_{i,k}$, use model variables K_i as a sum limit. One way to have proper fixed limits is to impose an upper-bound constraint $K_i \leq \overline{K}_i$ and use $\overline{K}_i$ as a perturbation limit. If $\overline{K}_i$ is significantly large, this implies that the optimal solution should place all extra steps beyond optimal $K_i$outside of the estimation horizon T (unless the constraint $\overline{T}_{i,k} \leq T$ is included) and thus eliminate their impact on the actual data fit.

A few examples illustrating flexibility and justifying the quadratic $X_{i,k}^T \cdot D \cdot X_{i,k}$ term:

Example 1. *Assume that for each i, the perturbation process $A_{i,t}$ is Poisson with constant out-of-state jump rates μ_i and deterministic state increments all equal to one. Taking the log of the joint probability density of the Poisson sojourn times implies that the term $X_{i,k}^T \cdot D \cdot X_{i,k} = \mu_i \cdot \left(\overline{T}_{i,k} - \underline{T}_{i,k}\right)$is linear. To ensure that there is no idle periods we need to include constraints $\overline{T}_{i,k} = \underline{T}_{i,k+1}$ and $\underline{T}_{i,k} \leq \overline{T}_{i,k}$. Also, to properly capture the deterministic state increments we need to impose an additional constraint $A_{i,k+1} - A_{i,k} = 1$.*

Example 2. *Assume that the jump state migration process is a Brownian motion. In this case $X_{i,k}^T \cdot D \cdot X_{i,k}$ includes expansion terms of the quadratic increment$(A_k - A_{k-1})^2$.*

Example 3. *Assume that the perturbations are rare and completely deterministic so that the zero value of $A_{i,t}$ represent a stable state and any deviations from it must be short and infrequent. A strong way to enforce such stability is to not to impose the constraint $\overline{T}_k = \underline{T}_{k+1}$ and choosing a relatively small value of $\overline{K}_i$ which is more appropriate for rare deterministic perturbations. Alternatively, a weak way to impose this stability is to assume that the rates of jumping out of the (positive) state A_k are* *state-dependent and linearly proportional to the value of A_k. That implies a mean-reverting bilinear term $A_k \cdot (\overline{T}_k - \underline{T}_k)$ in the objective function which penalizes for long durations of large perturbations.*

From the aforementioned examples, we have shown that (similar to the core continuous process monotonicity constraint $F_{t+1} \geq F_t$) perturbation structure specifics may require to enforce different kinds of linear equality or inequality constraint on both $A_{i,k}$ and $\underline{T}_{i,k}, \overline{T}_{i,k}$. Therefore, we include into our model formulation a generic polyhedral form of such constraints:

$$F_t \in F, X_{i,k} \in X_i \tag{3}$$

where F and each X_i is a polyhedron.

Finally, we need to reformulate the flat level constraint $A_{i,t} = \sum_{k=1}^{K_i} A_{i,k} \cdot I\left[\overline{T}_{i,k} \leq t < \underline{T}_{i,k}\right]$ in a more suitable form. For this, we define perturbations start & end binary indicator variables $\{\underline{w}_{i,k,t}\}_{i=1,k=1,t=1}^{N,K_i,T}$ and $\{\overline{w}_{i,k,t}\}_{i=1,k=1,t=1}^{N,K_i,T}$such that $\underline{w}_{i,k,t}, \overline{w}_{i,k,t} \in \{0,1\}$and use the following mixed-integer-quadratic form of the flat-level constraint:

$$\begin{aligned}
& A_{i,t} = \sum_{k=1}^{K_i} A_{i,k} \cdot \left(\underline{w}_{i,k,t} - \overline{w}_{i,k,t}\right) \\
& \underline{w}_{i,k,t+1} \geq \underline{w}_{i,k,t}; \overline{w}_{i,k,t+1} \geq \overline{w}_{i,k,t} \\
& \underline{w}_{i,k,t} \geq \overline{w}_{i,k,t} \\
& \overline{w}_{i,k,t} \geq \underline{w}_{i,k+1,t} \\
& \sum_{t=1}^{T} \left(\underline{w}_{i,k,t} - \overline{w}_{i,k,t}\right) = \overline{T}_{i,k} - \underline{T}_{i,k}.
\end{aligned} \tag{4}$$

This completes our MIP model definition so that the model includes quadratic objective (2) if the costs $c(K_i)$are assumed to be constant or linear, polyhedron constraints (3), quadratic constraints (4) and discrete binary constraints for perturbation start & end indicator variables$\{\underline{w}_{i,k,t}, \overline{w}_{i,k,t}\}$.

In this form, our problem is well-defined and is directly accepted by MIP solvers. However, we note that using $\{\underline{w}_{i,k,t}, \overline{w}_{i,k,t}\}$ explodes the solver time as T scales up ($\overline{K}_i$ should also be scaled up with time T if the data exhibits more perturbation on larger time horizons) and suffers from curse of dimensionality.

We have tried another form of our estimation problem that allows to use non-convex NLP solvers. This form can be obtained by using trapezoid approximations for perturbations step-function time profile. We redefine $\underline{w}_{i,k,t}, \overline{w}_{i,k,t} \in \{0,1\}$as indicator functions over time:

$$\begin{aligned}
I_{i,k,t} &= \min\{t, \overline{T}_{i,k}\} \\
J_{i,k,t} &= \max\{t, \underline{T}_{i,k}\}
\end{aligned} \tag{5}$$

and the flat level constraint as:

$$A_{i,t} = \sum_{k=1}^{K_i} \max[0, (A_{i,k} + |A_{i,k}|)/2 - M \cdot \left(J_{i,k,t} - I_{i,k,t}\right)] + \min[0, (A_{i,k} - |A_{i,k}|)/2 + M \cdot \left(J_{i,k,t} - I_{i,k,t}\right)]. \tag{6}$$

where the constant $M >> 1$ is the so called 'big-M' such that the slopes of the trapezoid (6) are steep enough to approximate a rectangular step function. This constraints are non-differentiable in certain points and thus either must be smoothed (e.g. via a sigmoid-like function) or a sub-gradient optimization must be used instead.

The NLP reformulation is faster but, as our numerical tests have indicated, does not produce robust estimates numerically as it is a non-convex problem with multiple optima and a flat near-optima profile of the objective function. The main reason for that is the fact that perturbation times $\overline{T}_{i,k}$ and$\underline{T}_{i,k}$ are variables rather than fixed parameters. If all these times were fixed than the MIP problem becomes a convex well-defined QP problem. However, estimation of those times is the heart of perturbation model. We exploit this idea in our Dynamic Programming approximation that solves such QP-problems in a recoursive matter while optimizing over perturbation time variables.

4 DYNAMIC PROGRAMMING ALGORITHM

As common to any dynamic programming problem, we go backwards in time starting from the end of the horizon $t = T$. At each point of time t, we solve the estimation problem on the $[t;T]$ horizon. Let us assume that we solved it and obtained optimal values for all model variables including the perturbation count $K_i^*(t)$ and the perturbation times $\overline{T}_{i,k}(t)$ and $\underline{T}_{i,k}(t)$ where $0 \le k \le K_i^*(t)$. We assume that the the perturbation times $\overline{T}_{i,k}(t)$ and $\underline{T}_{i,k}(t)$ are set and will not be changed for all subsequent solution steps for all $K_i^*(t)$ perturbations identified previously. Thus we remove the index (t)that indicates the estimation done at the step $t$and denote these times simply as $\overline{T}_{i,k}$and $\underline{T}_{i,k}$ where $0 \le k \le K_i^*(t)$

Now we move to the point $t-1$ and for each data series i we have an option of either:

1. Setting (the end mark) of a new perturbation here (i.e. setting $K_i^*(t-1) = K_i^*(t)+1$ and introducing $\overline{T}_{i,K_i^*(t-1)} = t-1$, $\underline{T}_{i,K_i^*(t-1)} = t-2$)
2. Continuing the previous perturbation (i.e. by setting $K_i^*(t-1) = K_i^*(t)$ and resetting $\underline{T}_{i,K_i^*(t-1)} = t-2$)

Note that if we model the case described in the Example 3 then we can introduce the third option of that is: terminating the previous perturbation and not resetting anything.

All these options vary in terms of objective value which must be evaluated (for every option) through a recourse problem defined on the horizon $[t-1;T]$.

First obvious cost difference is the extra perturbation for the Option 1 which comes with extra $c(K_i^*(t))$. Thus we remove the perturbation counting penalty $\sum_{i=1}^{N} c(K_i) \cdot K_i$ from the objective function and instead account the difference (between two options) $c(K_i^*(t))$ directly. However, we need to account for cost changes due to optimization over jump levels which are allowed to vary even if the perturbation times are fixed and thus can shapen the existing jumps and even completely shaving them off. Thus we define the recourse convex QP subproblems by using our MIP formulation on the the horizon $[t-1;T]$ with fixed $K_i = K_i^*(t)$ and perturbation times $\overline{T}_{i,k}$ and $\underline{T}_{i,k}$ where $0 \le k \le K_i^*(t)$. For each option, the flat-level constraint is reformulated in a slightly different way, that is provided below (for corresponding options):

1. We set the constraints $A_{i,u} = \sum_{k=1}^{K_i} A_{i,k} \cdot I\left[\overline{T}_{i,k} \le u < \underline{T}_{i,k}\right]$ for all $u \in [t;T]$and we also set the constraint $A_{i,t-1} = A_{i,K_i+1}$. Note that these are now linear constraints.
2. We set the constraints $A_{i,u} = \sum_{k=1}^{K_i} A_{i,k} \cdot I\left[\overline{T}_{i,k} \le u < \underline{T}_{i,k}\right]$ for all $u \in [t-1;T]$

The only mild extra assumption we impose for the sake of convexity is that the term $X_{i,k}^T \cdot D \cdot X_{i,k}$ is convex quadratic considering the fixed perturbation times. Thus, the overall QP subproblem is well-defined convex.

Note that we need to evaluate the recourse function for all possible enumerations of option 1 and option 2 between all Nseries which means that we need to do 2^N different calls of the QP subproblems. Then, we can choose the minimum value between them by exact search and fix the corresponding perturbation time changes. After that, we move to the point $t-2$ and repeat the procedure until the point $t = 1$ is reached. To complete the definition of our DP problem, we set the terminal condition as $K_i^*(T) = 0$ (in some cases it makes more sense to set $K_i^*(T) = 1, \overline{T}_{i,1} = T, \underline{T}_{i,1} = T-1$ as it may be better for our heuristic to start from being within a perturbation).

This DP approximation is only a heuristic for our global MIP model. However as we show in a numerical test in the next section, our DP heuristics performs rather accurate and robust for a difficult case where conventional continuous approach disregarding perturbations fails even though it is robust for large clustered continuous deviations. Moreover, every recourse subproblem call solves a well-defined convex QP problem and thus overall computational time scale polynomially with T . The solver time still scales exponentially with the number of sensors N. However, in realistic failure detection applications number of sensors for tracking a particular failure is usually small (less than 20).

5 NUMERICAL CASE ANALYSIS

For a fast and intuitive check of the quality of our DP approach, we designed and tested a small scale simulated case considered in this section. We simulate a $N = 3$ sensor data series over $T = 25$ time points horizon. As an underlying core failure process we take a completely deterministic process $F(t) = t$.

For each sensor we introduce two step-wise perturbations that are slightly not in-sync with each other:

Series 1 $A_{1,1} = -10; \underline{T}_{1,1} = 5; \overline{T}_{1,1} = 9; A_{1,2} = -15; \underline{T}_{1,2} = 15; \overline{T}_{1,1} = 21$
Series 2 $A_{2,1} = -14; \underline{T}_{2,1} = 4; \overline{T}_{2,1} = 9; A_{2,2} = -33; \underline{T}_{2,2} = 18; \overline{T}_{2,1} = 25$
Series 3 $A_{3,1} = -15; \underline{T}_{3,1} = 2; \overline{T}_{3,1} = 9; A_{3,2} = -35; \underline{T}_{3,2} = 17; \overline{T}_{3,1} = 23$

To each of the series, we added simulated i.i.d. Gaussian white noise with the standard deviation $q = 2.5$. The Figure 2 illustrates the simulated data with the core process $F(t) = t$ shown for each series:

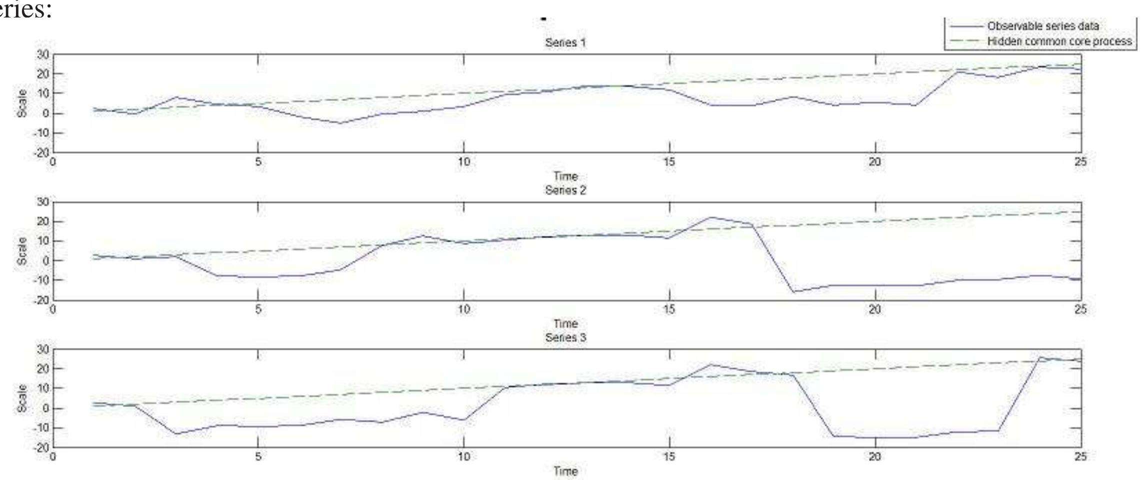

Figure 2: Actual observable series data.

Then we tested the regular QP estimation procedure (1) that ignored the perturbations. For numerical execution, we MATLAB with Mosek QP solver. The Figure 3 illustrates the estimation results of the core process $F(t)$:

As we can see, the regular QP estimation procedure (1) fails to produce an estimate that is even just close to the true form. Even the monotonicity constraint did not help to keep robustness since this procedure had to treat perturbed time periods in the same way as the other ones and thus could only react with delayed and decreased increments of its estimation for the process $F(t)$.

Next, we run our DP approximation assuming that the matrix $D$is all zeroes and costs $c(K_i)$ are constant and equal to 300. Also, we include the third option (as mentioned in the previous section) to the algorithm. The $F(t)$estimation results are given by the Figure 4 and the combined $F_t + A_{i,t}$ estimations for each series are given by the Figure 5:

The $F(t)$ estimation is very accurate except for the initial 25% of time where it is too conservative due to the fact that the actual $F(t)$ process has small scale comparable to the white noise level. As expected, the $A_{i,t}$ estimations are less accurate than F_t as each perturbation series was specific to the corresponding observations series and thus the perturbation estimation many degrees of freedom. Also, the DP heuristic partially contributed to that as there was no control over the total perturbations number used in the estimation. The perturbation estimation results produced are as follows:

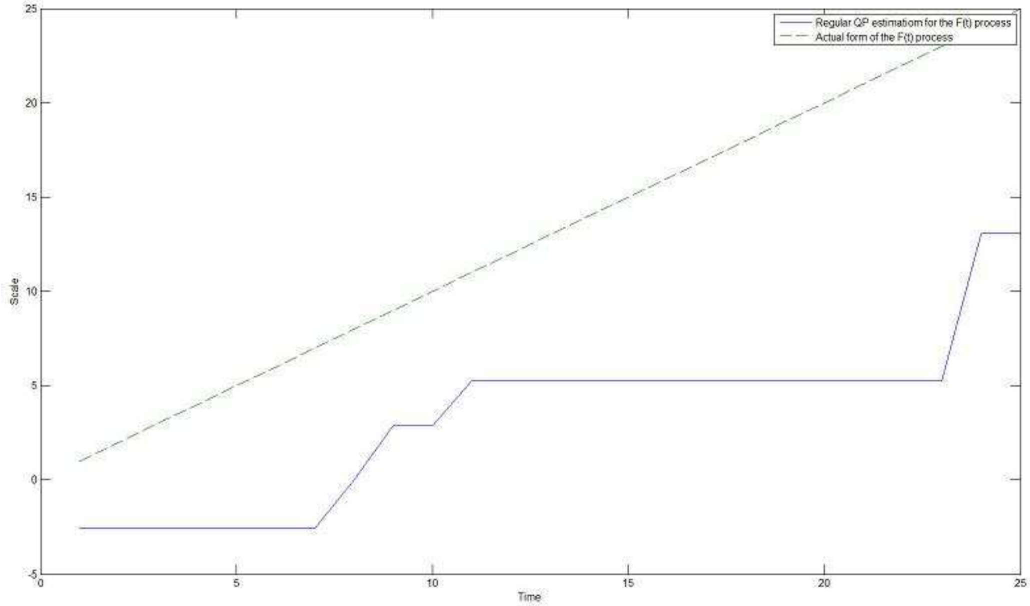

Figure 3: Regular QP estimation results.

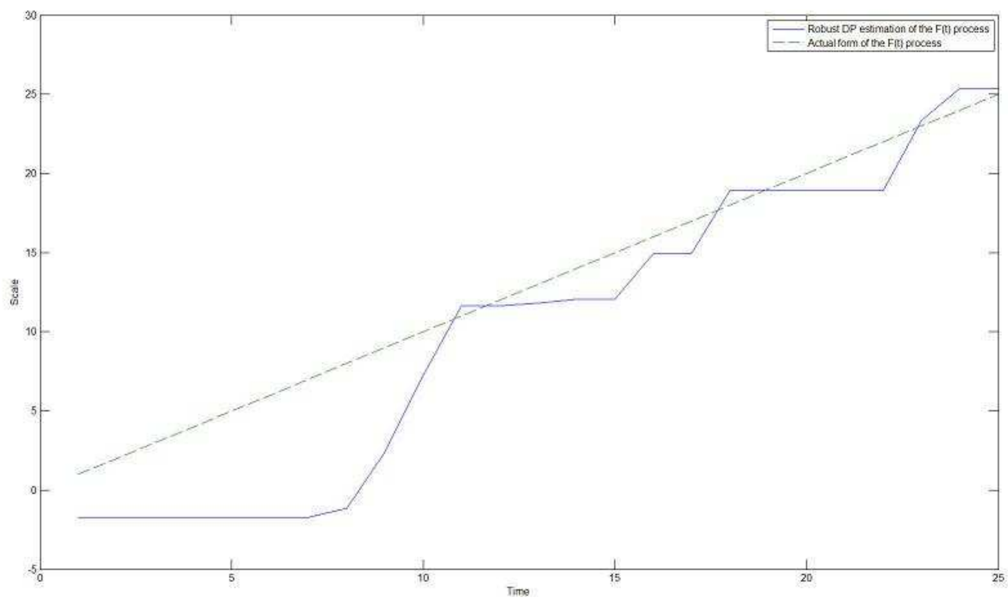

Figure 4: DP estimation results.

Series 1 $K_1 = 2; A_{1,1} = -1.5; \underline{T}_{1,1} = 5; \overline{T}_{1,1} = 13; A_{1,2} = -15.3; \underline{T}_{1,2} = 15; \overline{T}_{1,1} = 21$
Series 2 $K_2 = 3; A_{2,1} = -6.1; \underline{T}_{2,1} = 3; \overline{T}_{2,1} = 6; A_{2,2} = 1.3; \underline{T}_{2,2} = 7; \overline{T}_{2,2} = 9; A_{2,3} = -34.1; \underline{T}_{2,3} = 17; \overline{T}_{2,3} = 25$
Series 3 $K_3 = 2; A_{3,1} = -6.6; \underline{T}_{3,1} = 1; \overline{T}_{3,1} = 11; A_{3,2} = -33.2; \underline{T}_{3,2} = 17; \overline{T}_{3,1} = 23$

Our DP procedure very accurately identified the second perturbation for all time series since all the signals were strong enough to filter out & separate properly.

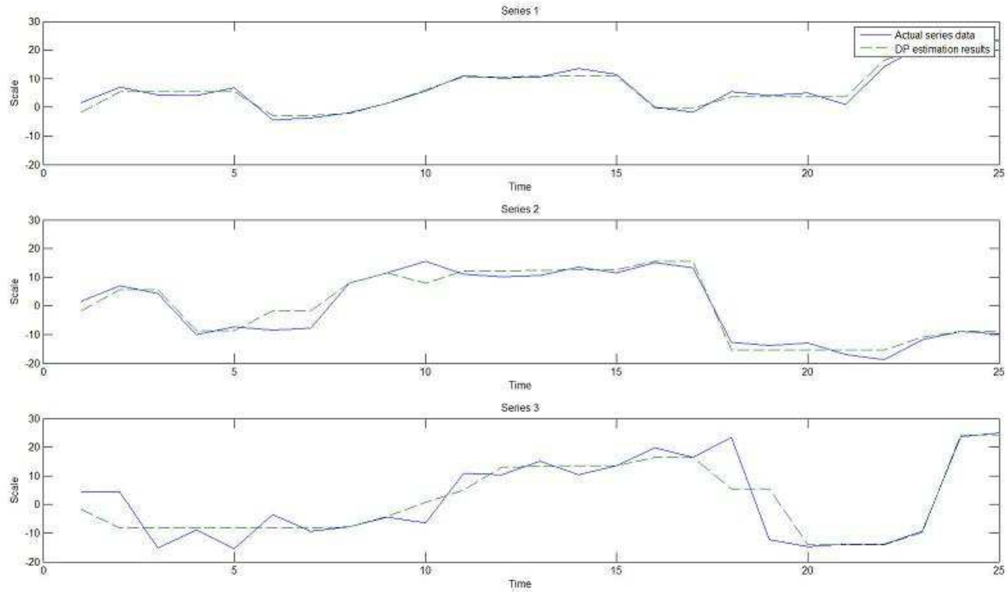

Figure 5: Final sensor signal DP estimations.

The total solution time for our DP heuristic was 28 seconds while our earlier MIP attempts did not produce a solution within 1 hour.

6 CONCLUSION

Overall, we have been satisfied with the estimation results & algorithm performance. We consider this DP approximation approach promising for practical applications of multivariate jump-diffusion processes estimation.

We plan to use this algorithm in one of our immediate works on stochastic state estimation for electric distribution networks. This application is particularly demanding for powerful and robust real-time estimation algorithms. It is a tightly coupled network problem and the estimation vector is highly constrained by network equations. This, in turn, allows to use fewer observation sensors but makes the problem harder. It also includes discrete switching states and controls and this is where we see the benefit of using our algorithm.

REFERENCES

Bandi, F., and T. Nguyen. 2003. On the functional estimation of jump–diffusion models. *Journal of Econometrics* 116(1):293–328.

Blackmore, L., S. Rajamanoharan, and B. Williams. 2008. Active estimation for jump markov linear systems. *IEEE Transactions on Automatic Control* 53(10):2223–2236.

Dufour, F., and P. Bertrand. 1994. The filtering problem for continuous-time linear systems with markovian switching coefficients. *Systems & Control Letters* 23(6):453–461.

Gertler, J. 1988. Survey of model-based failure detection and isolation in complex plants. *IEEE Control Systems Magazine* 8(6):3–11.

Jiang, G., and R. Owen. 2007. Estimating latent variables and jump diffusion models using high-frequency data. *Journal of Finacial Econometrics* 5(1):1–30.

Lee, S., and P. Mykland. 2008. Jumps in financial markets: A new nonparametric test and jump dynamics. *Review of Financial Studies* 21(6):2535–2563.

Mahmoud, M., and P. Shi. 2004. Optimal guaranteed cost filtering for markovian jump discrete-time systems. *Mathematical Problems in Engineering* 1:33–48.

Plageman, C., D. Fox, and W. Burgard. 2007. Efficient failure detection on mobile robots using particle filters with gaussian process proposals. *Proceedings of the 2007 International Joint Conference on Artificial Intelligence(IJCAI-2007)*:2185–2190.

Samar, S., A. Torzhkov, and A. Chakraborty. 2008. On-line monotonic fault diagnostics via recursive constrained quadratic programming. *Proceedings of the 2008 International Conference on Computational Intelligence for Modelling, Control and Automation(CIMCA2008)*:226–231.

Venkatasubramanian, V., R. Rengaswam, K. Yin, and S. Kavuri. 2003. A review of process fault detection and diagnosis: Part i:quantitative model-based methods quantitative model-based methods. *Computers & Chemical Engineering* 27(3):293–311.

Wolff, R. 1989. *Stochastic modeling and the theory of queues*. Englewood Cliffs, NJ: Prentice Hall.

Zhao, S., and G. Wei. 2003. Jump process for the trend estimation of time series. *Computational Statistics & Data Analysis* 42(1):219–241.

AUTHOR BIOGRAPHIES

ANDREY TORZHKOV is a Research Scientist at Siemens Corporate Research. He is an optimization expert in an R&D group that works in broad areas of condition monitoring and power systems analysis. His research interests are in networked systems optimization, stochastic estimation and smart grid technologies. He received his PhD in Operations Management from New York University. His e-mail address is <andrey.torzhkov@siemens.com>.

PUNEET SHARMA is a Research Scientist at Siemens Corporate Research. He is a comptutational fluid dynamics expert working on on medical informatics as well as engineering optimization problems. His research interests are in multi-physics systems of pertial differential equations and parameter estimation. He received his PhD in Industrial Engineering from University of Illionis, Urbana-Champaign. His e-mail address is <puneet.sharma@siemens.com>.

AMIT CHAKRABORTY is an R&D program manager within Siemens Corporate Research. His program mainly focuses on modeling and optimization techniques for advanced decision support. Some of the key technologies that he has worked on include Statistical Data Mining and Integration, Detection and Estimation, Convex Optimization and Game Theory. He has successfully led several R&D projects in energy management, condition monitoring and power systems analysis. He received his PhD in Electrical Engineering from Yale University. His e-mail address is <amit.chakraborty@siemens.com>.