Smooth Priorities

by
Francisco José P.A.V. Mendonça
Instituto Superior Técnico - Universidade Técnica de Lisboa
favm@mega.ist.utl.pt

Abstract - A new policy for multi-product, limited capacity production systems with uncertain demands is studied. Theoretical comparisons are made to the common policies of strict priorities and linear scaling. An optimizer based on IPA simulation is devised and results of practical comparison between smooth and strict priorities are presented. The structure of the cost function with smooth priorities is studied through function plots obtained from simulation.

I. Introduction

In the context of multi-product, limited capacity production systems with uncertain demands, the optimal policy has defied closed form formulations and shown to be excessively complex for numerical calculation. For this reason, it is frequent to use sub-optimal policies characterized by a small set of parameters which may then allow for simulation based optimization. The objective of this paper is to further study the smooth priorities policy. Within the sub-optimal policies normally used, decision based on a priority list is one of the most common. However, a strict priority policy makes difficult the satisfaction of demand for products of reduced priority, which tends to raise their levels of safety stock. Furthermore, determining the adequate list of priorities is a combinatorial problem of difficult resolution. Smooth priorities try to overcome both these shortfalls by allowing all products to access the production resources, weighted by their relative priorities, and by so doing transforming this into the continuous variable, nonlinear programming problem of determining those weights.

A great tutorial that describes the current state of the art in production systems is [1]. First results on the optimality of multi-echelon base-stock policies for uncapacitated multi-echelon production systems were presented in [2]. Optimality of base-stock policy for one product, single capacitated machine is proved for two different cases in [3] and [4]. A modified base-stock policy is shown optimal in the single product two-echelon capacitated case (when the machine that handles external demand has smaller capacity) in [5]. A practical approach that assumes a base-stock policy is applied to a single product multi-echelon capacitated production system is used to determine stability conditions in [6] and validate sensitivity analysis in [7]. The study of several common base-stock policies for several types of capacity allocation when applied to re-entrant multi-product capacitated systems in [8] first determines stability conditions, validates IPA and deduces a necessary optimality condition, then uses simulation based optimization to draw several practical conclusions. A first practical study of smooth priorities applied to multi-product capacitated systems is done in [9]. The optimality of base-stock policies in the multiple products, single machine case is approached in [10]. The existence of an optimal stationary base-stock policy, the optimality of the Weighted Balancing Rule in the case of equal cost parameters and symmetric demand distributions, and the asymptotic optimality of the Weighted Balancing Rule in the case of equal holding costs, high service levels, and symmetric demand is shown.

To obtain practical results, a simulation based optimization with gradient estimates provided by Infinitesimal Perturbation Analysis (IPA) (see [11]) is used. Simple substitution is used to prove that smooth priorities can encompass both Linear Scaling Rule (LSR) and strict priorities.

The general model used along with the LSR, strict priorities, and smooth priorities production equations, and relevant IPA equation are presented in II. The fact that LSR and strict priorities for two products are included in smooth priorities is proven in III. The simulation algorithm, the distribution generation techniques, the characteristics of the chosen optimization algorithm and the way statistical relevance of estimations is guaranteed are presented in IV. Practical cases studied, results obtained, and drawn conclusions are presented in V. Conclusions and future work are presented in VI.

II. Model and Base-Stock Policies

The general model used in this paper has M machines with finite capacity, P final products that are constituted by Fp phase products. Each phase product can be assigned to be produced in any machine without restriction (i.e., no serial or other structure is imposed) and can use up different amounts of capacity to produce (i.e., different loads), raw materials are always available (i.e., perfect delivery) and are used by the first phase product of a final product, demand is continuous and occurs on the last phase product of a final product after production. Product quantities are considered continuous and linear holding and backlog costs are used.
Figure 1 - Illustration of system model flexibility

Notation:

- $N$ - number of machines;
- $F$ - number of final products;
- $P_j$ - number of phase products of final product $p$;
- $M_m$ - set of phase products produce in machine $m$;
- $K_m$ - capacity of machine $m$;
- $I_{t,p}^{f,j}$ - inventory of phase product $f$ of final product $p$ in time period $t$;
- $E_{t,p}^{f,j}$ - echelon inventory of phase product $f$ of final product $p$ in time period $t$;
- $P_{t,p}^{f,j}$ - production of phase product $f$ of final product $p$ in time period $t$;
- $d_t^p$ - demand for final product $p$ at the end of time period $t$;
- $h_{t,p}^p$ - holding cost of phase product $f$ of final product $p$;
- $b^p$ - backlog cost of final product $p$;
- $\tau_{t,p}^f$ - load of phase product $f$ of final product $p$;
- $C_t^p$ - cost for final product $p$ in time period $t$;
- $\alpha_p$ - set of phase products produce in machine $m$;
- $z_{t,p}^f$ - echelon base-stock level of phase product $f$ of final product $p$;
- $\Delta_{t,p}^{f,j}$ - alternative (to $z_{t,p}^f$) set of variables that relate inventory between phases;
- $y_t^{p,f}$ - shortfall of phase product $f$ of final product $p$ in time period $t$;
- $f_t^{p,f}$ - production needs of phase product $f$ of final product $p$ in time period $t$;
- $\alpha_p$ - smooth priorities’ $\alpha$ parameter of phase product $f$ of final product $p$.

Equations that show how the system evolves from period to period:

$$I_{t+1}^{p,f} = I_{t}^{p,f} + P_{t}^{p,f} - P_{t+1}^{p,f}$$
$$E_{t+1}^{p,f} = E_{t}^{p,f} + P_{t}^{p,f} - d_t^p$$
$$E_t^{p,f} = \sum_{i=1}^{F} (I_t^{p,f})$$

Equations to determine the cost in each period:

$$C_t^p = -\min(I_t^{p,f},0) \times b^p + \sum_{f=1}^{F} (\max(I_t^{p,f},0) \times h_t^{p,f})$$
$$C_t^p = \sum_{p=1}^{P} (C_t^p)$$

Equations that define $\Delta_{t,p}^{f,j}$, $y_t^{p,f}$ and $f_t^{p,f}$:

$$\Delta_{t,p}^{f,j} = z_{t,p}^f - z_{t+1,p}^f$$
$$y_t^{p,f} = \max(z_{t,p}^f - E_{t}^{p,f}, 0)$$
$$f_t^{p,f} = \min(y_t^{p,f}, I_t^{p,f})$$

Description of base-stock policies and their production equations:

**Linear Scaling Rule (LSR)** - when production needs can not be satisfied for all products of a given machine production is linearly scaled to fit the machine capacity.

$$P_{t}^{p,f} = \begin{cases} 
  f_t^{p,f}, & \sum_{(i,j)\in M_+} f_{t}^{i,j} \leq K_m \\
  f_t^{p,f} \frac{K_m}{\sum_{(i,j)\in M_+} f_{t}^{i,j} \leq K_m}, & \sum_{(i,j)\in M_+} f_{t}^{i,j} > K_m 
\end{cases}$$

**Strict Priority** - each product has a priority assigned and fulfills production necessities by order of priority until machine capacity is fully used or there are no more products to produce. (in the following equation we consider products ordered by priority and $j=1$ means the product has the highest priority)

$$P_i^j = \min(f_t^{i,j}, \frac{K_m - \sum_{i=0}^{j-1} P_t^{i,j} \tau_j}{\tau_j}), \quad P_0^j = 0$$
Smooth Priorities - has two production phases where production is decided with LSR. Each product has a parameter \( \alpha \) that indicates the relative quantity of production needs that enter the first phase LSR. If there is remaining capacity after the first phase, \((1-\alpha)\) is the proportion that enters the second phase LSR.

\[
P_{i,t}^{p,f} = \alpha^{p,f} f_{i,t}^{p,f} \min\left\{ \frac{K_m}{\sum_{(i,j) \in M_u} \alpha^{f,j} f_{i,t}^{f,j} \tau^{f,j}}, 1 \right\}
\]

\[
P_{i,t}^{p,f} = (1-\alpha^{p,f}) f_{i,t}^{p,f} \min\left\{ \frac{K_m - \sum_{(i,j) \in M_u} P_{i,t}^{f,j} \tau^{f,j}}{(1-\alpha^{f,j}) f_{i,t}^{f,j} \tau^{f,j}}, 1 \right\}
\]

\[
P_{i,t}^{f} = P_{i,t}^{p,f} + P_{i,t}^{f,f}
\]

IPA equations (\( x \) is used to refer to any of the policy parameters)

System evolution:

\[
\frac{\delta I_{i}^{p,f}}{\delta x} = \frac{\delta d_{i}^{p,f}}{\delta x} + \frac{\delta P_{i}^{p,f}}{\delta x} - \frac{\delta P_{i}^{p,f+1}}{\delta x}, \quad \frac{\delta P_{i}^{p,f+1}}{\delta x} = \frac{\delta d_{i}^{p}}{\delta x} = 0
\]

Period costs:

\[
\frac{\delta C_{i}^{p}}{\delta x} = -[I_{i}^{p,f},<0] \times \frac{\delta I_{i}^{p,f}}{\delta x} b^{p} + \sum_{f=1}^{x} [I_{i}^{p,f},>0] \times \frac{\delta I_{i}^{p,f}}{\delta x} h^{p,f}
\]

\[
\frac{\delta C_{i}^{p}}{\delta x} = \sum_{j=1}^{x} \frac{\delta C_{j}^{p}}{\delta x}
\]

Production needs:

\[
\frac{\delta f_{i}^{p,f}}{\delta x} = \left\{ \begin{array}{ll}
0, & f_{i}^{p,f}=0 \\
\sum_{j=1}^{x} \frac{\delta \Delta^{i,j}}{\delta x} \sum_{j=1}^{x} \frac{\delta I_{i}^{p,f}}{\delta x}, & y_{i}^{p,f} < I_{i}^{p,f-1} \wedge f_{i}^{p,f} > 0, \quad I_{i}^{p,0} = \infty \quad \text{and} \quad \frac{\delta I_{i}^{p,0}}{\delta x} = 0 \\
\frac{\delta I_{i}^{p,f-1}}{\delta x}, & y_{i}^{p,f} > I_{i}^{p,f-1} \wedge f_{i}^{p,f} > 0
\end{array} \right.
\]

Strict Priorities:

\[
\frac{dP_{i}^{f}}{dx} = \sum_{i} f_{i}^{f} \tau^{f} < K_{m}
\]

Smooth Priorities:

\[
\frac{\delta P_{i}^{f,f}}{\delta x} = \left\{ \begin{array}{ll}
\frac{\delta \alpha^{f,f}}{\delta x} f_{i}^{p,f} + \alpha^{f,f} f_{i}^{p,f} & A_{i,t}^{f,f}(x), \\
A_{i,t}^{f,f}(x) \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j} & - \sum_{(i,j) \in M_u} A_{i,t}^{f,f}(x) \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j}, \\
K_{m} \left( \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j} \right)^{2} & \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j} > K_{m}
\end{array} \right.
\]

\[
\frac{\delta P_{i}^{p,f+1}}{\delta x} = \frac{\delta P_{i}^{p,f}}{\delta x} - \sum_{(i,j) \in M_u} P_{i,t}^{f,j} \tau^{f,j}, \\
K_{m} \sum_{(i,j) \in M_u} \left( \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j} \right) \frac{\delta P_{i}^{f,j}}{\delta x} = \sum_{(i,j) \in M_u} \alpha^{f,j} f_{i}^{f,j} \tau^{f,j} < K_{m} \wedge \sum_{(i,j) \in M_u} f_{i}^{f,j} \tau^{f,j} > K_{m}
\]

\[
\frac{\delta P_{i}^{f,f}}{\delta x} = \frac{\delta P_{i}^{p,f}}{\delta x} + \frac{\delta P_{i}^{p,f+1}}{\delta x} - \delta K_{m} - \sum_{(i,j) \in M_u} P_{i,t}^{f,j} \tau^{f,j}
\]

III. Theoretical Comparison of Policies

To show that the LSR policy is contained in the smooth priority policy the equality of both when smooth priority has all \( \alpha \) parameters equal is shown.

The smooth priority production equations can be expressed differently as a disjunction of all the possible cases which gets rid of the minimums as follows.
IV. Simulation based Optimization

have as many LSR phases as there are products in the machine the prof can be generalized for any amount of products.

When the highest strict priority product has

\( \alpha^{\mu, f} = \alpha \)

and simplifying we get

By doing \( \forall (p, f) \in M_\mu \) we get a null first phase giving the following equations

Joining the equally valued we finally arrive at

To show that in the two product case strict priorities are contained in smooth priorities the equality of both when the highest strict priority product has \( \alpha=1 \) and the lowest strict priority product has \( \alpha=0 \) is shown.

For the product 1 with \( \alpha=1 \) we get a null second phase of production and for the product 2 with \( \alpha=0 \) we get a null first phase giving the following equations

The equations for the strict priority case with only two products are the following

It can easily be seen that they are equivalent. It is also important to note that by extending smooth priorities to have as many LSR phases as there are products in the machine the prof can be generalized for any amount of products.

IV. Simulation based Optimization

The simulation algorithm and the techniques for the generation of distributions are adapted from [12]. The simulation algorithm is a general one but does assume that generated events are independent of state.

**Notation:**
- \( x \) - state
- \( e \) - event
- \( x^* = f(x, e) \) - model that describes state change when events occur
- \( t \) - time (continuous)
- \( (e, t^e) = g() \) - event generator that issues next event \( e \) and the time when it occurs \( t^e \)
- \( a(t, x, x^*, e) \) - evaluator that allows interesting quantities to be estimated during simulation
- \( A \) - set of evaluators
- \( x_0 \) - initial state of simulation
- \( t_0 \) - initial time of the simulation
- \( t_{\text{terminal}} \) - time we want to simulate until

**Description of Simulation Algorithm:**

1) \( x = x_0, \ t = t_0 \)
2) \( (e, t^e) = g() \)
3) if \( t^e > t_{\text{terminal}} \) end simulation
4) \( t = t^e \)
5) \( x^* = f(x, e) \)
6) for all \( a \) in \( A \): \( a(t, x, x^*, e) \)
7) \( x = x^* \)
8) return to 2
Distribution Generation Techniques
- Inverse Transform:
To generate a random variable $X$ with distribution $F(x) = P(X \leq x)$ (cumulative distribution function) from $U[0,1]$ (uniform probability distribution from 0 to 1 approximated by using rand() ) all that needs to be done is $X = F^{-1}(U)$ since the cumulative distribution function is always increasing and therefore the inverse always exist.
- Convolution:
To generate a random variable $X$ with distribution $F(x) = G_1(x) \ast \cdots \ast G_n(x)$ (cumulative distribution function) we use the fact that $X = \sum_{i=1}^{N} Y_i$ with $Y_i \sim G_i(x)$ has that distribution and so all that needs to be done is to sum the numbers $y_i$ generated from each distribution $G_i(x)$.

Choice of Optimization Algorithm
A review of several algorithms that can be consulted in [13] was made to choose the algorithm to use. Since the function we want to optimize (average cost for given policy parameters) is not convex, it is not always true that $(\nabla^2 f(x_k))^{-1}$ is positive definite, algorithms that use Hessian matrix approximations cannot be used. So, the Fletcher-Reeves method to determine the search direction was restarted with a gradient step every $n$ steps, to guarantee convergence. Since the optimization has constraints on the function's domain of the form $x_j > a \land x_i < b$ the projection into the domain is very simple to do and allows for restrictions to be enforced. Since the function is computationally expensive to compute the golden section algorithm was used for line searches.

The algorithm is thus described as follows:
1) $d_0 = -\nabla f(x_0)$
2) $d_k = \text{proj}(x_k + y d_k) - x_k$
3) $\alpha_k = \text{arg min}_{\alpha} f(x_k - \alpha d_k)$
4) $x_{k+1} = x_k + \alpha_k d_k$
5) if $\|x_{k+1} - x_k\| < \delta$ terminate
6) $d_{k+1} = -\nabla f(x_{k+1}) + \frac{\nabla f(x_k) + \nabla f(x_{k+1})}{\nabla f(x_k)} d_k$ and return to 2

Guaranteeing Statistical Relevance
To guarantee that the average cost value of the optimization result is within a small percentage interval with 95% confidence 45 different simulations for each set of policy parameters are run and the t-Student distribution with 44 degrees of freedom is used to test if 95% confidence is guaranteed if not the optimization is rerun with a higher time horizon for the simulation.

Note: For the case in study the state is a vector of inventory levels, $x^* = f(x,e)$ is a model of the factory with chosen production policy and the set of evaluators are an average cost evaluator and an average gradient evaluator based on the relevant IPA equations.

VI. Practical Results
First, the previous results of [9] are reproduced as closely as possible. This is done with the purpose of confirming the validity of previous results and validating the approach here proposed. Then a new simple case is studied for several costs. This serves the purpose of showing that smooth priorities do achieve upper bounds in terms of cost relative to the smooth priorities, as was established in section III, while showing that there is a choice of smoothing parameters that equals the behavior of strict priorities for only two products. A more complex case is studied and a parallel to the results of [14] is made. To finish, the structure of the cost function is studied and the extrapolation for the general case is attempted.

Comparison with Previous Results
The case is this:
- one machine, with capacity 25, produces two products
- product one has:
  average demand = 8
  inverse variance coefficient = 3
  backlog cost = 50
  holding cost = 10
- product two has:
  average demand = 12
  inverse variance coefficient = 1
  backlog cost = 20
  holding cost = 10

Note: For the case in study the state is a vector of inventory levels, $x^* = f(x,e)$ is a model of the factory with chosen production policy and the set of evaluators are an average cost evaluator and an average gradient evaluator based on the relevant IPA equations.

VI. Practical Results
First, the previous results of [9] are reproduced as closely as possible. This is done with the purpose of confirming the validity of previous results and validating the approach here proposed. Then a new simple case is studied for several costs. This serves the purpose of showing that smooth priorities do achieve upper bounds in terms of cost relative to the smooth priorities, as was established in section III, while showing that there is a choice of smoothing parameters that equals the behavior of strict priorities for only two products. A more complex case is studied and a parallel to the results of [14] is made. To finish, the structure of the cost function is studied and the extrapolation for the general case is attempted.

Comparison with Previous Results
The case is this:
- one machine, with capacity 25, produces two products
- product one has:
  average demand = 8
  inverse variance coefficient = 3
  backlog cost = 50
  holding cost = 10
- product two has:
  average demand = 12
  inverse variance coefficient = 1
  backlog cost = 20
  holding cost = 10

Note: For the case in study the state is a vector of inventory levels, $x^* = f(x,e)$ is a model of the factory with chosen production policy and the set of evaluators are an average cost evaluator and an average gradient evaluator based on the relevant IPA equations.
This is a case in which, according to [8], a strict priority policy would have better results than LSR and ESR (Equalized Shortfall Rule) since one of the two products has the lowest average demand, the highest backlog costs, and the least variance.

<table>
<thead>
<tr>
<th>Optimization Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>(\Delta^1)</td>
</tr>
<tr>
<td>10.513</td>
</tr>
<tr>
<td>11.9863</td>
</tr>
<tr>
<td>26.2372</td>
</tr>
</tbody>
</table>

Table 1 - Comparison between strict and smooth priorities

The results obtained by the optimizer for strict priority for 1st product, smooth priority and strict priority for 2nd product are presented in Table 1. Although the numerical results differ, the conclusions are the same, i.e., smooth priorities obtain better results than strict priorities.

**Different Costs**
The case is this:
- one machine, with capacity 100, produces two products
- product one has:
  - average demand = 40
  - exponential demand distribution
- product two has:
  - average demand = 40
  - exponential demand distribution
- the holding and backlog costs are varied

Results show that, for the cases where holding and backlog costs are the same for both products, smooth priorities has clearly better results and is equivalent to the use of the LSR. This was to be expected since there is no differentiating factor between the products.

In general, smooth priorities obtained better results than strict priorities but do converge to strict priorities in several of the cases where holding costs differ. For every case the order of the strict priorities with the best results correspond to order of the alphas of smooth priorities. As would be expected when holding or backlog costs rise for a product, smooth priorities get closer to strict priorities for that product.

This practically confirms that smooth priorities achieve better results than the equality with strict priorities proved theoretically (Section III). For the sake of space, we omit presenting the table with all the results here. These can be seen in [15].

**Complex System**
To explore the use of smooth priorities in a more complex structure the one illustrated in Figure 1 is used. The details are the following:
- 3 machines
- machine capacities: \(K_1=60\), \(K_2=100\), \(K_3=65\)
- 1st final product:
  - average demand = 30
  - exponential demand distribution
  - 4 phase products with holding costs: \(h^{1,1}=10\), \(h^{1,2}=15\), \(h^{1,3}=20\), \(h^{1,4}=25\)
  - backlog cost = 50
- 2nd final product:
  - average demand = 20
  - exponential demand distribution
  - 3 phase products with holding costs: \(h^{2,1}=10\), \(h^{2,2}=10\), \(h^{2,3}=10\)
  - backlog cost = 20
- the first machine produces:
  - phase product 1 of final product 1
  - phase product 3 of final product 2
- the second machine produces:
  - phase product 2 of final product 1
  - phase product 3 of final product 1
  - phase product 1 of final product 2
- the third machine produces:
  - phase product 4 of final product 1
  - phase product 2 of final product 2
The system was optimized with all the machines with smooth priorities and then with machine one with strict priority for product 2 and the remaining machines with smooth priorities. The results obtained (rounded) are shown in Table 2.

<table>
<thead>
<tr>
<th>Optimization Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta^{1,1}$</td>
</tr>
<tr>
<td>40.78</td>
</tr>
<tr>
<td>57.37</td>
</tr>
</tbody>
</table>

Table 2 - Smooth priorities vs strict priority of product 2 in machine 1

From the results we can see that the alpha parameters give priority to the phase products closer to the external demand. This brings to mind the "Last Buffer First Served" strict priorities policy that in [14] is the one that obtains the best results (in terms of mean cycle time), from those studied there, in a system with a re-entrant structure.

We should also take notice that applying strict priorities in machine 1 when the alpha parameters of smooth priorities give clear order of priority, but not strict priority, leads to an increase in the cost.

**Structure of the cost function**

Several graphs were produced, for both one of the simple and the more complex systems, to empirically study the cost function and one of those that presents the most complex curve for a single machine and two products is presented in Figure 2 and one of those that presents more complex curve for the system of Figure 1 is presented in Figure 3.
All graphs produced present a smooth cost function which makes it likely the same happens in general. Although the more complex graphs for a single machine do not present local minimums which permits us to speculate that the function is quasi-convex for that case, the more complex graph for the system of Figure 1 the same is not true as is proved by the existence of local minimums for the graph of Figure 3. Since from all the graphs this is the only one from a machine with more than two phase products but also with re-entrance it raises the question if this is due to the re-entrance or does it happen even for three products with no re-entrance.

Practical results are therefore coherent with theoretical expectations that the cost function is continuous but not with the expectations that it is quasi-convex, at least generally. This, unfortunately, means that there may be local minimums and so gradient based optimization may not obtain the optimal parameters.

VII. Conclusions and Future Work

A general system model that can encompass both serial and re-entrant systems was presented. Software was developed that implements the functioning of the systems, a general simulator and a general optimizer allowing for their optimization based on simulation.

Equations that express the function of the system where presented and where used to theoretically show that smooth priorities encompass the linear scaling rule in general and strict priorities in the case of two products.

With the use of the developed software, previous practical results where confirmed, the fact that smooth priorities have consistently better results than strict priorities was verified, the hypothesis that the ordering of the alphas of smooth priorities translate to an optimal ordering of the products for strict priorities was reinforced and the structure of the cost function was studied leading to the conclusion that it is most likely continuous but unfortunately is not quasi-convex in at least some cases.

Future work of interest is:
- Verifying whether the cost structure maintains the quasi-convex structure, shown for two products with no re-entrance, for more products and machines with no re-entrance.
- Practically explore the results of smooth priorities when compared with strict priorities for more than two products and more than one machine.
- Study the relevance of including bounds on local inventory in the various policies
- The creation of a graphical interface for the creation of the systems and a graphical program allowing the simulation and optimization of systems for several rules.
Bibliography


[14]: Lu, S.H. and Kumar, P.R., *Distributed Scheduling Based on Due-Dates and Buffer Priorities*, IEEE Trans. on Automatic Control 36(12):1406-1416, 1991