Topic one: Production line profit maximization subject to a production rate constraint

The profit maximization problem

$$
\begin{aligned}
\max_{\mathbf{N}} \quad J(\mathbf{N}) &= AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N}) \\
\text{s.t.} \quad P(\mathbf{N}) &\geq \hat{P}, \\
N_i &\geq N_{\min}, \forall i = 1, \cdots, k-1.
\end{aligned}
$$

where

$$
\begin{aligned}
P(\mathbf{N}) &= \text{production rate, parts/time unit} \\
\hat{P} &= \text{required production rate, parts/time unit} \\
A &= \text{profit coefficient, \$/part} \\
\bar{n}_i(\mathbf{N}) &= \text{average inventory of buffer } i, i = 1, \cdots, k-1 \\
b_i &= \text{buffer cost coefficient, \$/part/time unit} \\
c_i &= \text{inventory cost coefficient, \$/part/time unit}
\end{aligned}
$$

An example about the research goal

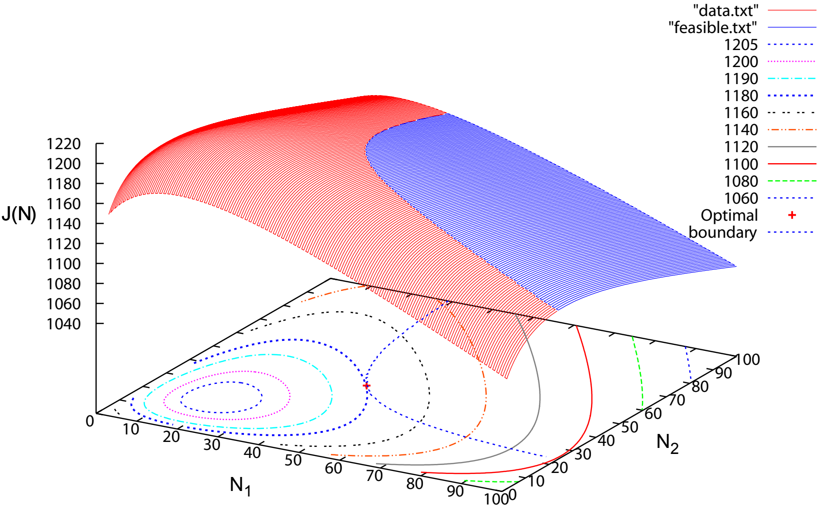


Figure 2: $J(\mathbf{N})$ vs. N_1 and N_2

Two problems

Original constrained problem

$$\max_{\mathbf{N}} \quad J(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N})$$

$$\text{s.t.} \quad P(\mathbf{N}) \geq \hat{P},$$

$$N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$

Simpler unconstrained problem (Schor's problem)

$$\max_{\mathbf{N}} \quad J(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N})$$

$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$

Data

$r_1 = .1, p_1 = .01, r_2 = .11, p_2 = .01, r_3 = .1, p_3 = .009, \hat{P} = .88$

Cost function

$J(\mathbf{N}) = 2000P(\mathbf{N}) - N_1 - N_2 - \bar{n}_1(\mathbf{N}) - \bar{n}_2(\mathbf{N})$

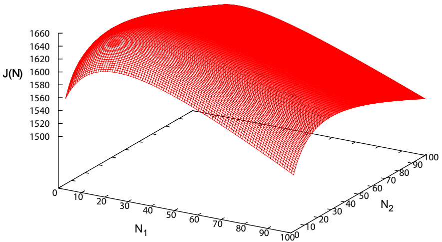


Figure 3: $J(\mathbf{N})$ vs. N_1 and N_2

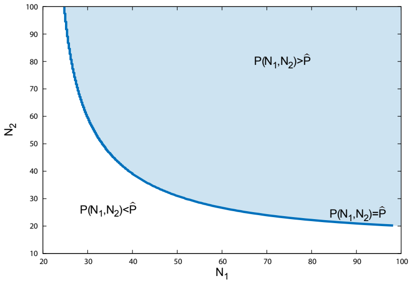


Figure 4: $P(\mathbf{N})$

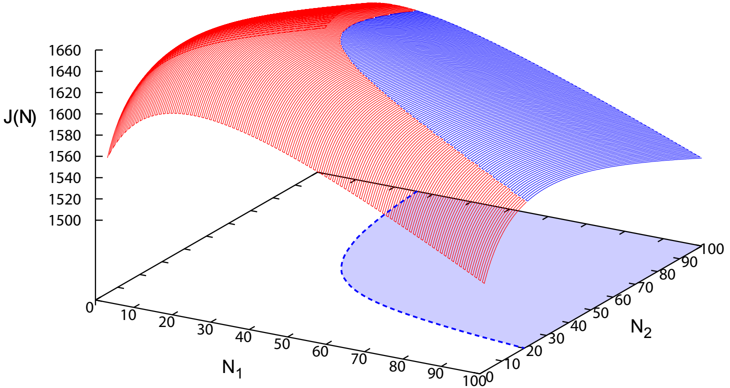


Figure 5: $J(\mathbf{N})$ vs. N_1 and N_2

Two cases

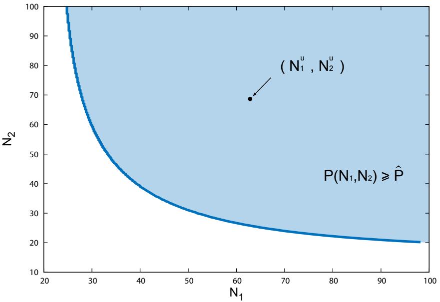
Case 1

The solution of the unconstrained problem is $\mathbf{N}^u$ s.t. $P(\mathbf{N}^u) \geq \hat{P}$. In this case, the solution of the constrained problem is the same as the solution of the unconstrained problem. We are done.

Unconstrained problem

$$\max_{\mathbf{N}} J(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N})$$

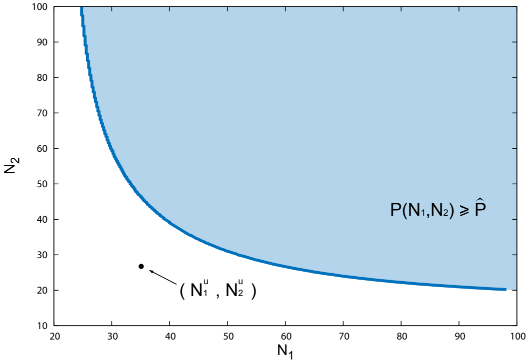
$$\text{s.t. } N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$



Two cases (continued)

Case 2

$\mathbf{N}^u$ satisfies $P(\mathbf{N}^u) < \hat{P}$. This is not the solution of the constrained problem.



Two cases (continued)

Case 2 (continued)

In this case, we consider the following unconstrained problem:

$$\max_{\mathbf{N}} \quad J(\mathbf{N}) = A' P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N})$$

$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$

in which A is replaced by A'. Let $\mathbf{N}^\star(A')$ be the solution to this problem and $P^\star(A') = P(\mathbf{N}^\star(A'))$.

The constrained problem

$$\begin{aligned} \max_{\mathbf{N}} \quad J(\mathbf{N}) &= A'P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N}) \\ \text{s.t.} \quad P(\mathbf{N}) &\geq \hat{P}, \\ N_i &\geq N_{\min}, \forall i = 1, \cdots, k-1. \end{aligned}$$

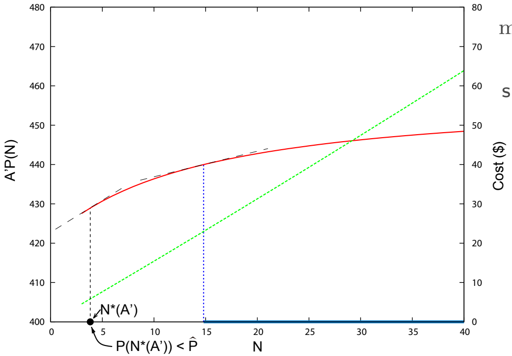
has the same solution for all A' in which the solution of the corresponding unconstrained problem

$$\begin{aligned} \max_{\mathbf{N}} \quad J(\mathbf{N}) &= A'P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N}) \\ \text{s.t.} \quad N_i &\geq N_{\min}, \forall i = 1, \cdots, k-1. \end{aligned}$$

has $P^\star(A') \leq \hat{P}$.

We claim

If the optimal solution of the unconstrained problem is not that of the constrained problem, then the solution of the constrained problem, $(N_1^\star, \cdots, N_{k-1}^\star)$, satisfies $P(N_1^\star, \cdots, N_{k-1}^\star) = \hat{P}$.



$$\max_{\mathbf{N}} J(N) = 500P(N) - N - \bar{n}(N)$$

$$\text{s.t. } P(N) \geq \hat{P}$$
$$N \geq N_{\min}$$

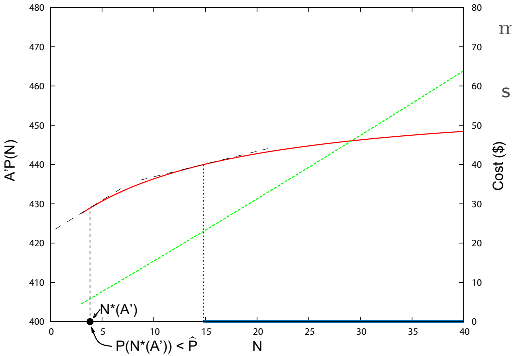
$$\max_{\mathbf{N}} J(\mathbf{N}) = 500\hat{P} - N - \bar{n}(N)$$

$$\text{s.t. } P(N) \geq \hat{P} \Rightarrow P(N) = \hat{P}$$
$$N \geq N_{\min}$$

We formally prove this by the Karush-Kuhn-Tucker (KKT) conditions of nonlinear programming.

We claim

If the optimal solution of the unconstrained problem is not that of the constrained problem, then the solution of the constrained problem, $(N_1^\star, \cdots, N_{k-1}^\star)$, satisfies $P(N_1^\star, \cdots, N_{k-1}^\star) = \hat{P}$.



$$\max_{\mathbf{N}} J(N) = 500P(N) - N - \bar{n}(N)$$

$$\text{s.t. } P(N) \geq \hat{P}$$
$$N \geq N_{\min}$$

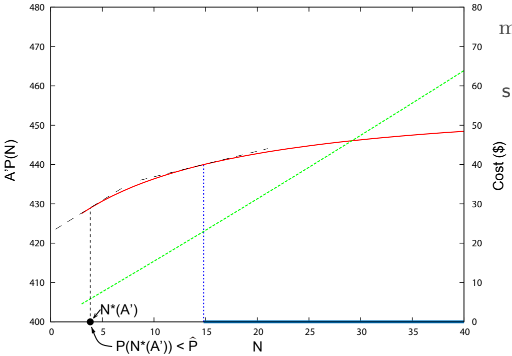
$$\max_{\mathbf{N}} J(\mathbf{N}) = 500\hat{P} - N - \bar{n}(N)$$

$$\text{s.t. } P(N) \geq \hat{P} \Rightarrow P(N) = \hat{P}$$
$$N \geq N_{\min}$$

We formally prove this by the Karush-Kuhn-Tucker (KKT) conditions of nonlinear programming.

We claim

If the optimal solution of the unconstrained problem is not that of the constrained problem, then the solution of the constrained problem, $(N_1^\star, \cdots, N_{k-1}^\star)$, satisfies $P(N_1^\star, \cdots, N_{k-1}^\star) = \hat{P}$.

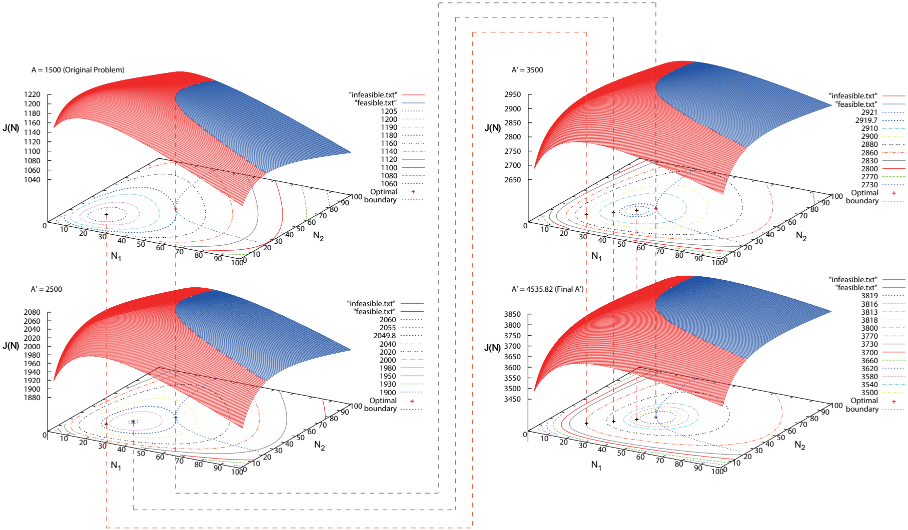


$$\begin{aligned} \max_{\mathbf{N}} J(N) &= 500P(N) - N - \bar{n}(N) \\ \text{s.t. } P(N) &\geq \hat{P} \\ N &\geq N_{\min} \end{aligned}$$

$$\begin{aligned} \max_{\mathbf{N}} J(\mathbf{N}) &= 500\hat{P} - N - \bar{n}(N) \\ \text{s.t. } P(N) &\geq \hat{P} \Rightarrow P(N) = \hat{P} \\ N &\geq N_{\min} \end{aligned}$$

We formally prove this by the Karush-Kuhn-Tucker (KKT) conditions of nonlinear programming.

Interpretation of the assertion



Let $x^\star$ be a local minimum of the problem

$$\begin{array}{ll} \min & f(x) \\ \text{s.t.} & h_1(x) = 0, \cdots, h_m(x) = 0, \\ & g_1(x) \leq 0, \cdots, g_r(x) \leq 0, \end{array}$$

where f, h_i, and g_j are continuously differentiable functions from $\Re^n$ to $\Re$. Then there exist unique Lagrange multipliers $\lambda_1^\star, \cdots, \lambda_m^\star$ and $\mu_1^\star, \cdots, \mu_r^\star$, satisfying the following conditions:

$$\nabla_x L(x^\star, \lambda^\star, \mu^\star) = 0,$$

$$\mu_j^\star \geq 0, j = 1, \cdots, r,$$

$$\mu_j^\star g_j(x^\star) = 0, j = 1, \cdots, r.$$

where $L(x, \lambda, \mu) = f(x) + \sum_{i=1}^m \lambda_i h_i(x) + \sum_{j=1}^r \mu_j g_j(x)$ is called the Lagrangian function.

Convert the constrained problem to minimization form

Minimization form

The constrained problem

$$\begin{aligned}
\min_{\mathbf{N}} \quad -J(\mathbf{N}) &= -AP(\mathbf{N}) + \sum_{i=1}^{k-1} b_i N_i + \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N}) \\
\text{s.t.} \quad \hat{P} - P(\mathbf{N}) &\leq 0 \\
N_{\min} - N_i &\leq 0, \forall i = 1, \cdots, k-1
\end{aligned}$$

We have argued that we treat N_i as continuous variables, and $P(N)$ and $J(N)$ as continuously differentiable functions.

The Slater constraint qualification for convex inequalities guarantees the existence of Lagrange multipliers for our problem. So, there exist unique Lagrange multipliers $\mu_i^\star, i = 0, \cdots, k-1$ for the constrained problem to satisfy the KKT conditions:

$$-\nabla J(\mathbf{N}^\star) + \mu_0^\star \nabla(\hat{P} - P(\mathbf{N}^\star)) + \sum_{i=1}^{k-1} \mu_i^\star \nabla(N_{\min} - N_i) = 0 \qquad (1)$$

or

$$-\begin{pmatrix} \frac{\partial J(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} - \mu_0^\star \begin{pmatrix} \frac{\partial P(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} - \mu_1^\star \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} - \cdots - \mu_{k-1}^\star \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \qquad (2)$$

and

$$\mu_i^\star \geq 0, \forall i = 0, \cdots, k-1, \quad (3)$$

$$\mu_0^\star(\hat{P} - P(\mathbf{N}^\star)) = 0, \quad (4)$$

$$\mu_i^\star(N_{\min} - N_i^\star) = 0, \forall i = 1, \cdots, k-1, \quad (5)$$

where $\mathbf{N}^\star$ is the optimal solution of the constrained problem. Assume that $N_i^\star > N_{\min}$ for all i. In this case, by equation (5), we know that $\mu_i^\star = 0, \forall i = 1, \cdots, k-1$.

The KKT conditions are simplified to

$$-\begin{pmatrix} \frac{\partial J(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} - \mu_0^\star \begin{pmatrix} \frac{\partial P(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \tag{6}$$

$$\mu_0^\star(\hat{P} - P(\mathbf{N}^\star)) = 0, \tag{7}$$

where $\mu_0^\star \geq 0$. Since $\mathbf{N}^\star$ is not the optimal solution of the unconstrained problem, $\nabla J(\mathbf{N}^\star) \neq 0$. Thus, $\mu_0^\star \neq 0$ since otherwise condition (6) would be violated. By condition (7), the optimal solution $\mathbf{N}^\star$ satisfies $P(\mathbf{N}^\star) = \hat{P}$.

The KKT conditions are simplified to

$$-\begin{pmatrix} \frac{\partial J(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial J(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} - \mu_0^\star \begin{pmatrix} \frac{\partial P(\mathbf{N}^\star)}{\partial N_1} \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_2} \\ \vdots \\ \frac{\partial P(\mathbf{N}^\star)}{\partial N_{k-1}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix},$$

$$\mu_0^\star(\hat{P} - P(\mathbf{N}^\star)) = 0,$$

In addition, conditions (6) and (7) reveal how we could find $\mu_0^\star$ and $\mathbf{N}^\star$. For every $\mu_0^\star$, condition (6) determines $\mathbf{N}^\star$ since there are $k-1$ equations and $k-1$ unknowns. Therefore, we can think of $\mathbf{N}^\star = \mathbf{N}^\star(\mu_0^\star)$. We search for a value of $\mu_0^\star$ such that $P(\mathbf{N}^\star(\mu_0^\star)) = \hat{P}$. As we indicate in the following, this is exactly what the algorithm does.

Replacing $\mu_0^\star$ by $\mu_0 > 0$ in constraint (6) gives

$$-\begin{pmatrix} \frac{\partial J(\mathbf{N}^c)}{\partial N_1} \\ \frac{\partial J(\mathbf{N}^c)}{\partial N_2} \\ \vdots \\ \frac{\partial J(\mathbf{N}^c)}{\partial N_{k-1}} \end{pmatrix} - \mu_0 \begin{pmatrix} \frac{\partial P(\mathbf{N}^c)}{\partial N_1} \\ \frac{\partial P(\mathbf{N}^c)}{\partial N_2} \\ \vdots \\ \frac{\partial P(\mathbf{N}^c)}{\partial N_{k-1}} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad (8)$$

where $\mathbf{N}^c$ is the unique solution of (8). Note that $\mathbf{N}^c$ is the solution of the following optimization problem:

$$\begin{aligned} \min_{\mathbf{N}} \quad & -\bar{J}(\mathbf{N}) = -J(\mathbf{N}) + \mu_0(\hat{P} - P(\mathbf{N})) \\ \text{s.t.} \quad & N_{\min} - N_i \leq 0, \forall i = 1, \cdots, k-1. \end{aligned} \quad (9)$$

The problem above is equivalent to

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = J(\mathbf{N}) - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \le 0, \forall i = 1, \cdots, k-1. \tag{10}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \le 0, \forall i = 1, \cdots, k-1. \tag{11}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = (A + \mu_0)P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i$$
$$\text{s.t.} \quad N_i \ge N_{\min}, \forall i = 1, \cdots, k-1. \tag{12}$$

The problem above is equivalent to

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = J(\mathbf{N}) - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \leq 0, \forall i = 1, \cdots, k-1. \tag{10}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \leq 0, \forall i = 1, \cdots, k-1. \tag{11}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = (A + \mu_0)P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i$$
$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1. \tag{12}$$

The problem above is equivalent to

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = J(\mathbf{N}) - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \leq 0, \forall i = 1, \cdots, k-1. \tag{10}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i - \mu_0(\hat{P} - P(\mathbf{N}))$$
$$\text{s.t.} \quad N_{\min} - N_i \leq 0, \forall i = 1, \cdots, k-1. \tag{11}$$

or

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = (A + \mu_0)P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i$$
$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1. \tag{12}$$

or, finally,

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = A' P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i \tag{13}$$

$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$

where $A' = A + \mu_0$. This is exactly the unconstrained problem, and $\mathbf{N}^c$ is its optimal solution. Note that $\mu_0 > 0$ indicates that $A' > A$.

In addition, the KKT conditions indicate that the optimal solution of the constrained problem $\mathbf{N}^\star$ satisfies $P(\mathbf{N}^\star) = \hat{P}$. This means that, for every $A' > A$ (or $\mu_0 > 0$), we can find the corresponding optimal solution $\mathbf{N}^c$ satisfying condition (8) by solving problem (13). We need to find the A' such that the solution to problem (13), denoted as $\mathbf{N}^\star(A')$, satisfies $P(\mathbf{N}^\star(A')) = \hat{P}$.

or, finally,

$$\max_{\mathbf{N}} \quad \bar{J}(\mathbf{N}) = A'P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i \tag{13}$$

$$\text{s.t.} \quad N_i \geq N_{\min}, \forall i = 1, \cdots, k-1.$$

where $A' = A + \mu_0$. This is exactly the unconstrained problem, and $\mathbf{N}^c$ is its optimal solution. Note that $\mu_0 > 0$ indicates that $A' > A$.

In addition, the KKT conditions indicate that the optimal solution of the constrained problem $\mathbf{N}^\star$ satisfies $P(\mathbf{N}^\star) = \hat{P}$. This means that, for every $A' > A$ (or $\mu_0 > 0$), we can find the corresponding optimal solution $\mathbf{N}^c$ satisfying condition (8) by solving problem (13). We need to find the A' such that the solution to problem (13), denoted as $\mathbf{N}^\star(A')$, satisfies $P(\mathbf{N}^\star(A')) = \hat{P}$.

Then, $\mu_0 = A' - A$ and $\mathbf{N}^\star(A')$ satisfy conditions (6) and (7):

$$-\nabla J(\mathbf{N}^\star(A')) + \mu_0^\star \nabla(\hat{P} - P(\mathbf{N}^\star(A'))) = 0,$$

$$\mu_0^\star(\hat{P} - P(\mathbf{N}^\star(A'))) = 0.$$

Hence, $\mu_0^\star = A' - A$ is exactly the Lagrange multiplier satisfying the KKT conditions of the constrained problem, and $\mathbf{N}^\star = \mathbf{N}^\star(A')$ is the optimal solution of the constrained problem.

Consequently, solving the constrained problem through our algorithm is essentially finding the unique Lagrange multipliers and optimal solution of the problem.

Then, $\mu_0 = A' - A$ and $\mathbf{N}^\star(A')$ satisfy conditions (6) and (7):

$$-\nabla J(\mathbf{N}^\star(A')) + \mu_0^\star \nabla(\hat{P} - P(\mathbf{N}^\star(A'))) = 0,$$

$$\mu_0^\star(\hat{P} - P(\mathbf{N}^\star(A'))) = 0.$$

Hence, $\mu_0^\star = A' - A$ is exactly the Lagrange multiplier satisfying the KKT conditions of the constrained problem, and $\mathbf{N}^\star = \mathbf{N}^\star(A')$ is the optimal solution of the constrained problem.

Consequently, solving the constrained problem through our algorithm is essentially finding the unique Lagrange multipliers and optimal solution of the problem.

Solve unconstrained problem

Solve, by a gradient method, the unconstrained problem for fixed A'

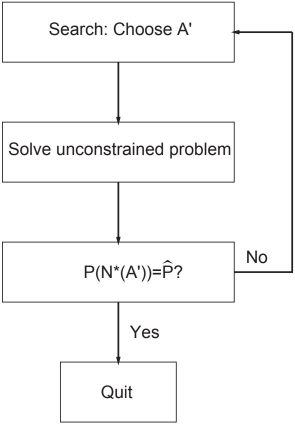
$$\max_{\mathbf{N}} \quad J(\mathbf{N}) \quad = \quad A'P(\mathbf{N}) - \sum_{i=1}^{k-1} b_i N_i - \sum_{i=1}^{k-1} c_i \bar{n}_i(\mathbf{N})$$

$$\text{s.t.} \qquad N_i \quad \geq \quad N_{\min}, \forall i = 1, \cdots, k-1.$$

Search

Do a one-dimensional search on $A' > A$ to find A' such that the solution of the unconstrained problem, $\mathbf{N}^\star(A')$, satisfies

$$P(\mathbf{N}^\star(A')) = \hat{P}.$$



Numerical experiment outline

- Experiments on short lines.
- Experiments on long lines.
- Computation speed.

Method we use to check the algorithm

$\hat{P}$ surface search in $(N_1, \cdots, N_{k-1})$ space. All buffer size allocations, $\mathbf{N}$, such that $P(\mathbf{N}) = \hat{P}$ compose the $\hat{P}$ surface.

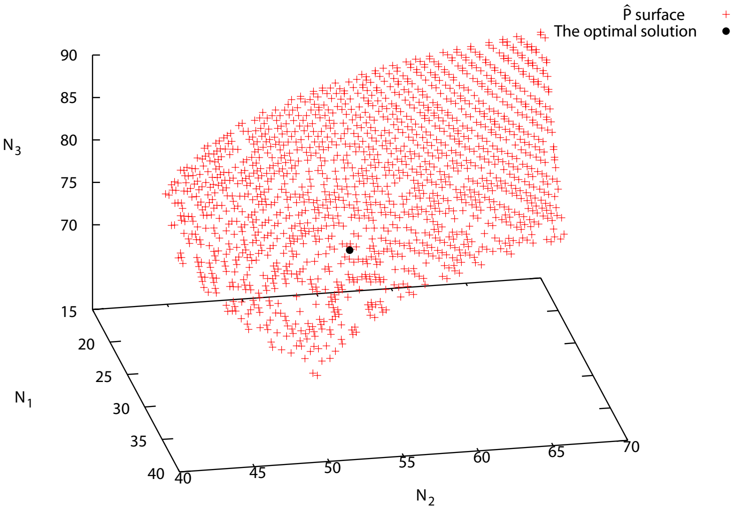


Figure 6: $\hat{P}$ Surface search

- Line parameters: $\hat{P} = .88$

machine	M_1	M_2	M_3	M_4	M_5
r	.11	.12	.10	.09	.10
p	.008	.01	.01	.01	.01

- Machine 4 is the least reliable machine (bottleneck) of the line.

- Cost function

$$J(\mathbf{N}) = 2500P(\mathbf{N}) - \sum_{i=1}^{4} N_i - \sum_{i=1}^{4} \bar{n}_i(\mathbf{N})$$

Results

- Optimal solutions

	$\hat{P}$ Surface Search	The algorithm	Error	Rounded $N^\star$
Prod. rate	.8800	.8800		.8800
$N_1^\star$	28.85	28.8570	0.02%	29.0000
$N_2^\star$	58.46	58.5694	0.19%	59.0000
$N_3^\star$	92.98	92.9068	0.08%	93.0000
$N_4^\star$	87.39	87.4415	0.06%	87.0000
$\bar{n}_1$	19.0682	19.0726	0.02%	19.1791
$\bar{n}_2$	34.3084	34.3835	0.23%	34.7289
$\bar{n}_3$	48.7200	48.6981	0.04%	48.9123
$\bar{n}_4$	31.9894	32.0063	0.05%	31.9485
Profit ($)	1798.2	1798.1	0.006%	1797.4000

- The maximal error is 0.23% and appears in $\bar{n}_2$.
- Computer time for this experiment is 2.69 seconds.

- Line parameters: $\hat{P} = .88$

machine	M_1	M_2	M_3	M_4	M_5	M_6
r	.11	.12	.10	.09	.10	.11
p	.008	.01	.01	.01	.01	.01

machine	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}
r	.10	.11	.12	.10	.12	.09
p	.009	.01	.009	.008	.01	.009

- Cost function

$$J(\mathbf{N}) = 6000P(\mathbf{N}) - \sum_{i=1}^{11} N_i - \sum_{i=1}^{11} \bar{n}_i(\mathbf{N})$$

Results

- Optimal solutions, buffer sizes:

	$\hat{P}$ Surface Search	The algorithm	Error	Rounded $N^\star$
Prod. rate	.8800	.8800		.8799
$N_1^\star$	29.10	29.1769	0.26%	29.0000
$N_2^\star$	59.20	59.2830	0.14%	59.0000
$N_3^\star$	97.80	97.7980	0.002%	98.0000
$N_4^\star$	107.50	107.4176	0.08%	107.0000
$N_5^\star$	84.50	84.4804	0.02%	84.0000
$N_6^\star$	70.80	70.6892	0.17%	71.0000
$N_7^\star$	63.10	63.1893	0.14%	63.0000
$N_8^\star$	53.10	52.9274	0.33%	53.0000
$N_9^\star$	47.20	47.2232	0.05%	47.0000
$N_{10}^\star$	47.90	47.7967	0.22%	48.0000
$N_{11}^\star$	48.80	48.7716	0.06%	49.0000

Results (continued)

- Optimal solutions, average inventories:

	$\hat{P}$ Surface Search	The algorithm	Error	Rounded $N^\star$
$\bar{n}_1$	19.2388	19.2986	0.31%	19.1979
$\bar{n}_2$	34.9561	35.0423	0.25%	34.8194
$\bar{n}_3$	52.5423	52.6032	0.12%	52.6833
$\bar{n}_4$	45.1528	45.1840	0.07%	45.0835
$\bar{n}_5$	34.4289	34.4770	0.14%	34.2790
$\bar{n}_6$	30.7073	30.7048	0.01%	30.8229
$\bar{n}_7$	28.0446	28.1299	0.30%	28.0902
$\bar{n}_8$	21.5666	21.5438	0.11%	21.5932
$\bar{n}_9$	21.5059	21.5442	0.18%	21.4299
$\bar{n}_{10}$	22.6756	22.6496	0.11%	22.7303
$\bar{n}_{11}$	20.8692	20.8615	0.04%	20.9613
Profit ($)	4239.3	4239.2	0.002%	4239.5000

- Computer time is 91.47 seconds.

Consider a 4-machine 3-buffer line with constraints $\hat{P} = .87$. In addition, $A = 2000$ and all b_i and c_i are 1.

machine	M_1	M_2	M_3	M_4
r_{i1}	.10	.12	.10	.20
p_{i1}	.01	.008	.01	.007
r_{i2}	–	.20	–	.16
p_{i2}	–	.005	–	.004

	$\hat{P}$ Surf. Search	The algorithm	Error
$P(\mathbf{N}^\star)$	.8699	.8699	
$N_1^\star$	29.8600	29.9930	0.45%
$N_2^\star$	38.2200	38.0206	0.52%
$N_3^\star$	20.6800	20.7616	0.39%
$\bar{n}_1$	17.2779	17.3674	0.52%
$\bar{n}_2$	17.2602	17.1792	0.47%
$\bar{n}_3$	6.1996	6.2121	0.20%
Profit ($)	1610.3000	1610.3000	0.00%

Consider a 4-machine 3-buffer line with constraints $\hat{P} = .87$. In addition, $A = 2000$ and all b_i and c_i are 1.

machine	M_1	M_2	M_3	M_4
μ_i	1.0	1.02	1.0	1.0
r_{i1}	.10	.12	.10	.20
p_{i1}	.01	.008	.01	.012
r_{i2}	–	.20	–	.16
p_{i2}	–	.005	–	.006

	$P^\star$ Surf. Search	The algorithm	Error
$P(\mathbf{N}^\star)$	.8699	.8700	
$N_1^\star$	27.7200	27.9042	0.66%
$N_2^\star$	38.7900	38.9281	0.34%
$N_3^\star$	34.0700	34.1574	0.26%
$\bar{n}_1$	15.4288	15.5313	0.66%
$\bar{n}_2$	19.8787	19.9711	0.46%
$\bar{n}_3$	13.8937	13.9426	0.35%
Profit ($)	1590.0000	1589.7000	0.02%

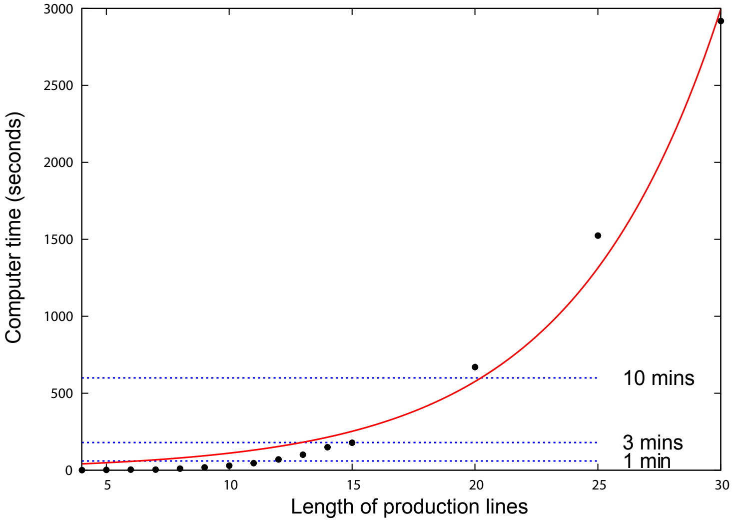
Experiment

- Run the algorithm for a series of experiments for lines having identical machines to see how fast the algorithm could optimize longer lines.
- Length of the line varies from 4 machines to 30 machines.
- Machine parameters are $p = .01$ and $r = .1$.
- In all cases, the feasible production rate is $\hat{P} = .88$.
- The objective function is

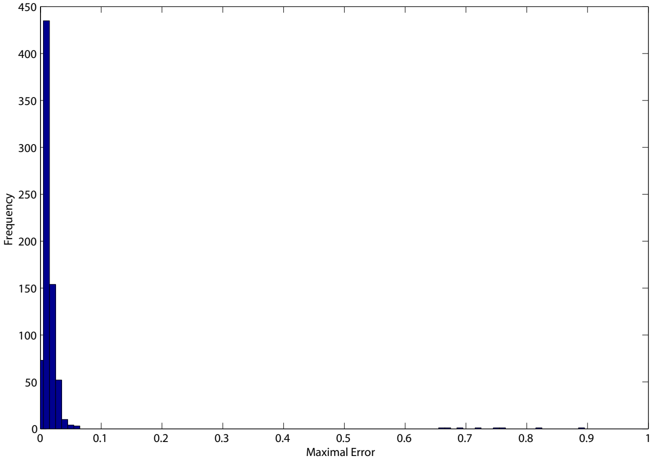
$$J(\mathbf{N}) = AP(\mathbf{N}) - \sum_{i=1}^{k-1} N_i - \sum_{i=1}^{k-1} \bar{n}_i(\mathbf{N}).$$

where $A = 500k$ for the line of length k.

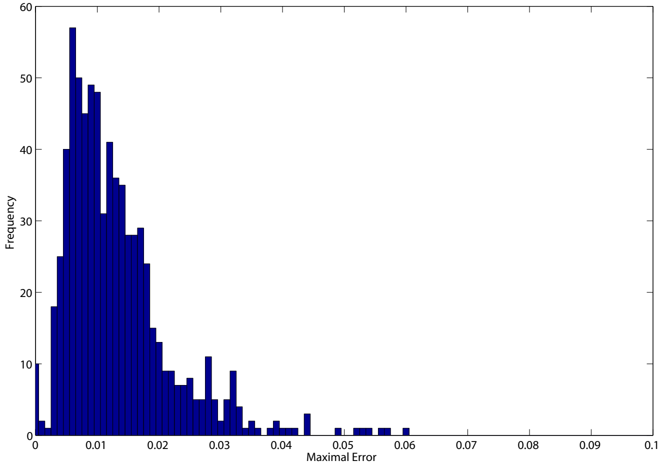
Computation speed



We run the algorithm on 739 randomly generated 4-machine 3-buffer lines. 98.92% of these experiments have a maximal error less than 6%.



Taking a closer look at those 98.92% experiments, we find a more accurate distribution of the maximal error. We find that, out of the total 739 experiments, 83.90% of them have a maximal error less than 2%.



2.852 Manufacturing Systems Analysis
Spring 2010