Annals of Operations Research 138, 203–222, 2005

Optimal Planning of a Multi-Station System with Sojourn Time Constraints

MARCOS SINGER* singer@faceapuc.cl
PATRICIO DONOSO
JOSÉ LUIS NOGUER
Escuela de Administración, Pontificia Universidad Católica de Chile, Vicuña Mackenna 4860, Macul, Santiago Chile

Abstract. This paper studies a dynamic production system where multiple products must visit stations where inventories are constrained by maximum and minimum sojourn times with neither negative flow nor backlog being allowed. A resource availability constraint limits the aggregate throughput of the stations. The objective is to minimize the sum of flow and inventory cost. The problem is broken down into several single-product serial systems that serve as subroutines of a Lagrangian relaxation routine. This model is implemented in a spreadsheet so that it can be used by the officials of a Chilean institution for planning the operations and defining the optimal allocation of resources.

Keywords: dynamic system, sojourn times, Lagrangian relaxation, operations planning, resource allocation

Introduction

This paper examines a system of several stations that sequentially process a number of products during many time periods. The cost from flows and inventories is assumed to be linear with no restriction in sign, so if negative, it corresponds to a benefit. The objective is to find the production plan defining how much inventory of each type of product should be transferred between consecutive stations at every time period, in order to minimize the total cost over time. There is an upper and a lower bound for the sojourn time that the average unit of inventory can spend within a station. In this context we understand "sojourn time" as the time-difference between the moment a given unit leaves a particular station and the instant it leaves the previous one. For the case of the first station, the sojourn time corresponds to the time elapsed between the moment the unit arrives in the system and the instant it leaves the first station. This period, considered in our model as a whole, may include a waiting time until service, a delay due to a relocation, the actual processing time, etc. Both flows and inventories must take non-negative values, hence there is no backward flow or inventory backlog. There is a cost of transferring products in terms of the utilization of limited resources shared by all the stations, restraining the total flow in the system. Cost and constraint parameters are specific to product types, stations and

* Corresponding author.

time periods, and are assumed to be certainly known, thus the model has a deterministic behavior.

Very recent research has been focused on multi-station systems, since their general structure can model manufacturing environments, service operations, and supply chains. In the first case, the stations may represent machines transforming goods and materials. Sojourn time constraints may arise when such conversion cannot be done too fast or too slow. For instance, an oven may require a minimum time to heat a given material, or otherwise could burn its content up. A food processor may hold inventory for at most a given time period in order to preserve its quality. In service operations the stations may represent work centers performing different tasks for the clients, so the inventory corresponds to work-in-process queuing up. Sojourn times may have a lower bound because doing the tasks in a rush could jeopardize their quality. For instance, some stages of a bank loan require a minimum time given the need for examining the client's information and obtaining back-office approvals. There can also exist maximum sojourn times when the service quality degrades, or the system's general effectiveness decreases with excessive delays. For instance, if clients end up waiting for too long, they can get impatient, dissatisfied or even leave. Section 5 provides a detailed example of sojourn times constraints within the judicial system. Given the strict time limits defined by the law for some stages of a civil action, if those limits are not complied the trial can be automatically dismissed. Finally, when multi-station systems model supply chains, every stage represents an autonomous company selling goods and services to one another. The inventory, work-in-process and the sojourn time constraints have similar interpretations to the ones described for the manufacturing and the service systems.

The motivation for studying these type of systems is of great importance to practitioners, as shown by a survey performed by Berry, Evans, and Naim (1998) about the pipeline information used by managers in the UK industry. The serial systems can also serve as building blocks for more complex models, as shown by Rosling (1998), who proves that assembly systems can be modeled as flow lines when there are no set-up costs. According to Yang, Yan, and Sethi (1999), algorithms for deterministic models can serve as a base of methods for stochastic flow-line models. This is also true for discrete systems such as the stochastic flow shop and job shop, for which Singer (2000) derives a scheduling method based on the deterministic shifting bottleneck heuristic by Pinedo and Singer (1999).

Yang, Yan, and Sethi (1999) study a deterministic system where multiple products visit the stations in a serial manner, minimizing the inventory and backlog cost. Assuming constant product demands, non-decreasing inventory costs along the flow and throughput constraints, they propose an algorithm of linear complexity that reaches a steady state behavior with zero inventories. Tan and Yeralan (1997) analyze a single-product serial system where machines are subject to random breakdowns and repair downtime. They decompose the problem into single-station subproblems in which inventories are independent of their adjacent stations. Through an iterative method, inventories are modified until they reach a balanced flow solution. Bradley and Arntzen (1999) study the problem of production planning and tradeoff between capacity and inventory in a deterministic framework. They use a dynamic model where several products follow serial or more complex paths across the system. For each station they define inventory constraints, variable cost parameters, capacity, depreciation cost and other parameters, and propose a mixed-integer program maximizing the return on operational assets. They apply the model to study an electronic equipment manufacturer and a school supplies producer. Regarding the analysis of sojourn times in flow lines, Ganesh (1998) derives the delay distribution of the entire system given general assumptions, a relevant result to communication networks that must guarantee some service quality. Altiok (2000) models the operation at a bulk material handling port where vessels go through several stages of delays. He proposes sojourn time bounds, as well as approximations, a significant performance measure for the port service quality.

Within the supply chains setup, Chen (1998) analyses multi-echelon systems with deterministic demand and backlogging, restricting the production plan to be stationary, i.e., each station receives a constant batch in regular time intervals, with both the batch and the interval being station-specific. Although the requirement of being stationary may be 70% below optimal, with randomly generated examples he reaches an effectiveness of 99%. Graves, Kletter, and Hetzel (1998) decompose a dynamic system into a number of single-stage, single-product subproblems facing uncertain demand, forecast and revised periodically. They analyze their production plan, inventory and capacity, and use those results as building blocks for a more complex network, such as a film manufacturer.

This paper is organized as follows: Section 1 presents a linear programming formulation for the multi-station system, similar to the model by Yang, Yan, and Sethi (1999) but without backlogging or any other restriction on the flow structure and the inventory costs. It shares most of the assumptions made by Bradley and Arntzen (1999), such as deterministic production yields and prices, constant variable costs independent of capacity, and availability of already processed products to successive stations in the same time period. Our formulation is relaxed in Section 2 in order to decompose it into several subproblems. Unlike Tan and Yeralan (1997) and Graves, Kletter, and Hetzel (1998), who define single-station subproblems, we generate single-product multi-station systems for which we introduce an optimal dynamic programming algorithm of linear complexity. In Section 3 an iterative Lagrangian relaxation routine is proposed for solving the original problem using the optimal single-station subroutine. In Section 4 the randomly generated benchmark instances used to test the method are introduced. We present the computational results showing a rapid convergence to close-to-optimal solutions with relatively low constraint violations. Having validated the method, we apply it to the case of the Chilean General Treasury. Section 5 outlines its operation and explain how the multi-station model, implemented in a spreadsheet, can assist in production planning and resource allocation. Section 6 presents our main conclusions.

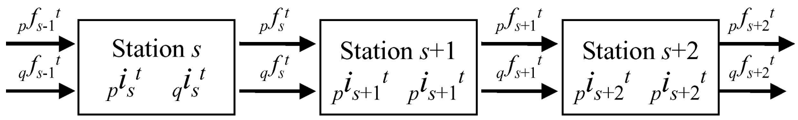

Figure 1. Production system with two products and three stations.

1. Process representation and optimization problem

We analyze a system that processes P types of products sequentially, visiting S stations during a time horizon divided into T periods of equal duration. Each station has only one predecessor or upstream station, as suggested by figure 1. Define the decision variable ${}_pf_s^t$ as the flow measured in units per time period of product p transferred from station s to station $s+1$ during time period t, with $p = 1, \ldots, P; s = 1, \ldots, S$ and $t = 1, \ldots, T$. Assume ${}_pf_s^t \geq 0$ so there is no backward flow, and consider the input flow ${}_pf_0^t$ of the first station a parameter of the problem, for $p = 1, \ldots, P$ and $t = 1, \ldots, T$. Define the decision variable ${}_pi_s^t$ as the inventory measured in units of product p in station s at the end of time period t. Assume ${}_pi_s^t \geq 0$ so there is no backlogging, and consider the inventory ${}_pi_s^0$ at time zero a parameter of the problem, for $p = 1, \ldots, P$ and $s = 1, \ldots, S$.

Define ${}_p\mathrm{F}_s^t$ as the unit cost related to the flow and ${}_p\mathrm{I}_s^t$ as the unit cost related to inventory, thus the flow and the inventory cost due to product p at station s during time period t is obtained by ${}_p\mathrm{F}_s^t \cdot {}_pf_s^t$ and ${}_p\mathrm{I}_s^t \cdot {}_pi_s^t$ respectively. Neither ${}_p\mathrm{F}_s^t$ nor ${}_p\mathrm{I}_s^t$ have a sign restriction; if negative, it corresponds to a benefit. At every time period it must be decided how much inventory must be transferred between consecutive stations in order to minimize total costs over time defined by:

$$\sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} {}_p\mathrm{F}_s^t \cdot {}_pf_s^t + \sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} {}_p\mathrm{I}_s^t \cdot {}_pi_s^t \tag{1}$$

Besides the non-negativity restrictions for variables ${}_pf_s^t$ and ${}_pi_s^t$, there are three families of constraints applied to each product, station and time period: maximum sojourn time, minimum sojourn time and flow balance. The maximum and minimum sojourn time constraints place an upper and a lower bound for the time the average unit can stay at any station. Suppose, for simplicity, that ${}_pf_s^t$ is the best predictor of ${}_pf_s^{t'}$ for $(t + {}_pi_s^t/{}_pf_s^t) \geq t' > t$, i.e., the flow in the near future of product p from station s is likely to be similar to the current one. If so, as reviewed by Dallery and Gershwin (1992) and Govil and Fu (1999), the expected sojourn time of the average unit of product p arriving in station s at the beginning of time period t is $({}_pi_s^{t-1}/{}_pf_s^t)$, while the expected sojourn time of the unit arriving at the end of time period t is $({}_pi_s^t/{}_pf_s^t)$. Therefore, the expected sojourn time of the average unit of product p arriving at station s throughout time period t is calculated as follows:

$$\frac{\left({}_p i_s^{t-1}/{}_p f_s^t\right) + \left({}_p i_s^t/{}_p f_s^t\right)}{2} = \frac{{}_p i_s^{t-1} + {}_p i_s^t}{2 \cdot {}_p f_s^t}$$
$$(p = 1, \ldots, P; s = 1, \ldots, S \text{ and } t = 1, \ldots, T)$$

The expression above is a sort of average of the cycle times calculated by Little's law: cycle time = inventory ÷ throughput. If the throughput ${}_p f_s^t$ remained as a constant between time t and time $(t + {}_p i_s^t/{}_p f_s^t)$, then the cycle time of the average unit arriving during such period would be exact, because the last arrived unit would leave the station in at most ${}_p i_s^t/{}_p f_s^t$ time periods. However, in this dynamic system the flow may change along time, so our approximation of the sojourn times critically relies on the already stated assumption that ${}_p f_s^t$ is a good estimation of the throughput shortly after t.

The upper bound of the expected sojourn time can be interpreted as a maximum makespan constraint for the average work order. Defining ${}_p \mathrm{C}_s^t$ as the maximum sojourn time the average unit of product p can stay at station s during time period t,

$$\frac{{}_p i_s^{t-1} + {}_p i_s^t}{2_p f_s^t} \leq {}_p \mathrm{C}_s^t \rightarrow {}_p i_s^t + {}_p i_s^{t-1} - 2 \cdot {}_p \mathrm{C}_s^t \cdot {}_p f_s^t \leq 0$$
$$(p = 1, \ldots, P; s = 1, \ldots, S \text{ and } t = 1, \ldots, T) \quad (2)$$

The lower bound of the expected sojourn time can be interpreted as a feasibility constraint forcing every work order to remain a minimum amount of time before being released. Defining ${}_p \mathrm{c}_s^t$, with ${}_p \mathrm{c}_s^t \leq {}_p \mathrm{C}_s^t$, as the minimum sojourn time the average unit of product p can stay at station s during time period t,

$$\frac{{}_p i_s^{t-1} + {}_p i_s^t}{2 \cdot {}_p f_s^t} \geq {}_p \mathrm{c}_s^t \rightarrow -{}_p i_s^t - {}_p i_s^{t-1} + 2 \cdot {}_p \mathrm{c}_s^t \cdot {}_p f_s^t \leq 0$$
$$(p = 1, \ldots, P; s = 1, \ldots, S \text{ and } t = 1, \ldots, T) \quad (3)$$

The flow balance equations links the inventory ${}_p i_s^t$, the incoming flow ${}_p f_{s-1}^t$ from the supplier, the outgoing flow ${}_p f_s^t$ and the inventory ${}_p i_s^{t-1}$ at the end of period $t - 1$ as follows:

$$-{}_p i_s^t + {}_p i_s^{t-1} + {}_p f_{s-1}^t - {}_p f_s^t = 0 \quad (p = 1, \ldots, P; s = 1, \ldots, S; t = 1, \ldots, T) \quad (4)$$

There is an additional set of constraints regarding the availability of different types of resources at every time period, limiting the aggregate throughput of the stations according to their utilization of resources such as labor or energy. Define the parameter ${}^r\mathrm{M}^t$ as the maximum availability of resource r at time t, with $r = 1, \ldots, R$ and $t = 1, \ldots, T$, and define parameter ${}_p^r\mathrm{U}_s^t$ indicating how much product p uses resource r when processed by station s at time t. The maximum resource constraint can be expressed as:

$$\sum_{p=1}^{P} \sum_{s=1}^{S} {}_p^r\mathrm{U}_s^t \cdot {}_p f_s^t \leq {}^r\mathrm{M}^t \quad (r = 1, \ldots, R \text{ and } t = 1, \ldots, T) \quad (5)$$

Although we are not defining an upper bound for the capacity of each station to hold inventory, expression (2) does not allow the inventory to grow too much in a given station s' once the flow is fixed. As expression (5) may be defined for the same s' by making ${}^r_p\mathrm{U}^t_s = 0$ for all $s \neq s'$, these two constraints combined can restrict the inventory to a maximum level.

In summary, the optimization problem consists of finding ${}_p f^t_s \geq 0$ and ${}_p i^t_s \geq 0$ that minimize (1) subject to (2),

(3), (4) and (5). Since all the expressions are linear and the parameters are known for certain, it is possible to solve the problem using a standard linear programming package. As an alternative, we develop a Lagrangian relaxation algorithm, generally used for non-linear formulations, given its computational efficiency later verified in Section 4. The use of non-linear techniques such as logarithmic barrier, inverse barrier, and exponential penalty for solving linear problems has been widely studied in the literature, including Auslander, Cominetti, and Haddou (1997) and Van Maaren and Terlaky (1997). Those techniques solve a sequence of subproblems defined in terms of a "step" δ, converging to the original problem as δ approaches zero. As their performance critically depends on how efficiently each subproblem can be solved, we focus on these and postpone the study of the modification of δ to future research.

2. Optimal algorithm for the single-product model

In the minimization problem stated above, only the resource availability constraint in expression (5) is coupling the different products. Therefore, by relaxing such constraint the problem can be decomposed into P single-product subproblems, for which an optimal dynamic programming algorithm based on the Karush Kuhn Tucker (KKT) condition will be proposed here. At this point, also the non-negativity constraints are relaxed, but they will be later reconsidered by rescaling the input data. Define the vector of flows $\vec{f} = (f^1_1, \ldots, f^T_1, \ldots, f^1_S, \ldots, f^T_S)$ and the vector of inventories $\vec{i} = (i^1_1, \ldots, i^T_1, \ldots, i^1_S, \ldots, i^T_S)$, both with $S \cdot T$ components that correspond to the variables defined in the previous section, but disregarding the sub-index p since there is only one product under consideration. Define $\mathbf{F} = (\mathrm{F}^1_1, \ldots, \mathrm{F}^T_1, \ldots, \mathrm{F}^1_S, \ldots, \mathrm{F}^T_S)$ and $\mathbf{I} = (\mathrm{I}^1_1, \ldots, \mathrm{I}^T_1, \ldots, \mathrm{I}^1_S, \ldots, \mathrm{I}^T_S)$ analogously. The following linear program minimizes the flow and inventory cost of S stations over T periods:

$$
\begin{array}{rlll}
\text{Minimize:} & \mathbf{F}\vec{f} + \mathbf{I}\vec{i} & & (6) \\
\text{Subject to:} & \check{\mathbf{C}}^f \vec{f} + \check{\mathbf{C}}^i \vec{i} \leq 0 & \text{Maximum sojourn time constraint} & (7) \\
 & \hat{\mathrm{C}}^f \vec{f} + \hat{\mathrm{C}}^i \vec{i} \leq 0 & \text{Minimum sojourn time constraint} & (8) \\
 & \mathbf{A}^f \vec{f} + \mathbf{A}^i \vec{i} = 0 & \text{Flow balance} & (9)
\end{array}
$$

The objective function (6) accounts for the variable cost due to flow and inventory. The maximum sojourn time constraints (7) come from expression (2), hence $\check{\mathbf{C}}^f$ and $\check{\mathbf{C}}^i$ are both $(S \cdot T) \times (S \cdot T)$ matrices with:

$$\check{\mathbf{C}}^f_{i,j} = \begin{cases} -2\mathrm{C}^t_s & \text{for } i = j \\ 0 & \text{otherwise} \end{cases} \quad \check{\mathbf{C}}^i_{i,j} = \begin{cases} 1 & \text{for } i = j \\ 1 & \text{for } i = j+1 \quad \text{and} \quad i \neq nT \text{ with } n = 1, \ldots, S \\ 0 & \text{otherwise} \end{cases}$$

For instance, with $T = 2$ and $S = 2$ matrices $\check{\mathbf{C}}^f$ and $\check{\mathbf{C}}^i$ are:

$$\check{\mathbf{C}}^f = \begin{bmatrix} -2\mathrm{C}^1_1 & 0 & 0 & 0 \\ 0 & -2\mathrm{C}^2_1 & 0 & 0 \\ 0 & 0 & -2\mathrm{C}^1_2 & 0 \\ 0 & 0 & 0 & -2\mathrm{C}^2_2 \end{bmatrix} \quad \check{\mathbf{C}}^i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}$$

The minimum sojourn time constraints (8) come from the expression (3), so $\hat{\mathbf{C}}^f$ and $\hat{\mathbf{C}}^i$ are also $(S \cdot T) \times (S \cdot T)$ matrices with:

$$\hat{\mathbf{C}}^f_{i,j} = \begin{cases} 2\mathrm{c}^t_s & \text{for } i = j \\ 0 & \text{otherwise} \end{cases} \quad \hat{\mathbf{C}}^i_{i,j} = \begin{cases} -1 & \text{for } i = j \\ -1 & \text{for } i = j+1 \quad \text{and} \quad i \neq nT \text{ with } n = 1, \ldots, S \\ 0 & \text{otherwise} \end{cases}$$

For instance, with $T = 2$ and $S = 2$, matrices $\hat{\mathbf{C}}^f$ and $\hat{\mathbf{C}}^i$ are:

$$\hat{\mathbf{C}}^f = \begin{bmatrix} 2\mathrm{c}^1_1 & 0 & 0 & 0 \\ 0 & 2\mathrm{c}^2_1 & 0 & 0 \\ 0 & 0 & 2\mathrm{c}^1_2 & 0 \\ 0 & 0 & 0 & 2\mathrm{c}^2_2 \end{bmatrix} \quad \hat{\mathbf{C}}^i = \begin{bmatrix} -1 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & -1 \end{bmatrix}$$

The flow balance constraints in (9) come from expression (4), hence $\mathbf{A}^f$ and $\mathbf{A}^i$ are $(S \cdot T) \times (S \cdot T)$ matrices, with:

$$\mathbf{A}^f_{ij} = \begin{cases} -1 & \text{for } i = j \\ 1 & \text{for } i = j + T \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{A}^i_{ij} = \begin{cases} -1 & \text{for } i = j \\ 1 & \text{for } i = j + 1 \quad \text{and} \quad i \neq nT \quad \text{with } n = 1, \ldots, S \\ 0 & \text{otherwise} \end{cases}$$

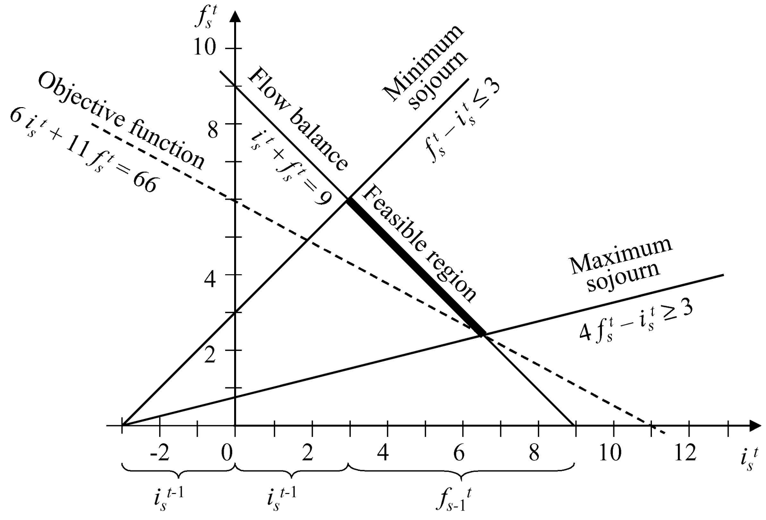

Figure 2. Feasible region in dimensions f_s^t and i_s^t.

For instance, for $T = 2$ and $S = 2$, $\mathbf{A}^f$ and $\mathbf{A}^i$ are:

$$\mathbf{A}^f = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad \mathbf{A}^i = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}$$

In order to fix ideas, figure 2 shows the feasible region of constraints (7), (8) and (9) in the dimensions f_s^t and i_s^t, assuming $f_{s-1}^t = 6$, $i_s^{t-1} = 3$, $\mathrm{F}_s^t = 11$ and $\mathrm{I}_s^t = 6$.

According to KKT, a necessary condition for a given point to be an optimal solution for a linear program is that the vector defining the objective function can be expressed as a linear combination of the active constraints' normal vectors. If the point is also feasible, then the condition becomes sufficient. In order to apply this property, define $\boldsymbol{\eta} = (\eta_1^1, \ldots, \eta_1^T, \ldots, \eta_S^1, \ldots, \eta_S^T)$ where η_s^t is the shadow price of the maximum sojourn time constraint for station s on period t. Since such constraint is in the form "$\leq$", η_s^t must be non-negative. Define $\boldsymbol{\mu} = (\mu_1^1, \ldots, \mu_1^T, \ldots, \mu_S^1, \ldots, \mu_S^T)$ where μ_s^t is the shadow price of the minimum sojourn time constraint for station s on period t, with μ_s^t also being non-negative. Define $\boldsymbol{\iota} = (\iota_1^1, \ldots, \iota_1^T, \ldots, \iota_S^1, \ldots, \iota_S^T)$ with ι_s^t as the shadow price of the flow balance equation for station s on period t. As the inventory constraint corresponds to an equality, ι_s^t has no restriction of sign. The KKT necessary condition can be met if the following system has a solution:

$$\begin{aligned} &\boldsymbol{\eta}\check{\mathbf{C}}^f + \boldsymbol{\mu}\widehat{\mathbf{C}}^f + \boldsymbol{\iota}\boldsymbol{A}^f = \mathbf{F} \\ &\boldsymbol{\eta}\check{\mathbf{C}}^i + \boldsymbol{\mu}\widehat{\mathbf{C}}^i + \iota\mathbf{A}^i = \mathbf{I} \\ &\eta \geq \mathbf{0}; \quad \mu \geq \mathbf{0}. \end{aligned} \tag{10}$$

The interpretation of the shadow prices η_s^t, μ_s^t and ι_s^t determines how binding its constraint is, and therefore how profitable it would be to relax it. If $\eta_s^t > 0$ then the maximum sojourn time constraint is active for station s on period t, hence it would be convenient to increase such time. If $\mu_s^t > 0$ then the minimum sojourn time constraint is active, so it would be convenient to reduce such time. If $\iota_s^t > 0$ then station s is overloaded with inventory at time period t, therefore one should try to discard inventory by selling it as unfinished product. Alternatively, if $\iota_s^t < 0$ then the inventory at station s is scarce at time period t, so one should try to increase it, possibly buying it from external providers.

The system of equations (10) is solved using backward induction, starting from the last station S and the last time period T. The result is the following algorithm for defining the dual variables, where $\mu_s^{T+1} \equiv \eta_s^{T+1} \equiv \iota_s^{T+1} \equiv \iota_{S+1}^t \equiv 0$ by definition.

Algorithm 1 (Dual Variable, Single-Product)

FOR $s = S \rightarrow 1$ REPEAT
  FOR $t = T \rightarrow 1$ REPEAT

$$\text{IF } \mathrm{F}_s^t + \iota_{s+1}^t + \mu_s^{t+1} \geq \mathrm{I}_s^t + \iota_s^{t+1} + \eta_s^{t+1} \text{ THEN} \tag{11}$$

$$\mu_s^t = 0$$

$$\eta_s^t = \frac{\mathrm{F}_s^t + \iota_{s+1}^t + \mu_s^{t+1} - \left(\mathrm{I}_s^t + \iota_s^{t+1} + \eta_s^{t+1}\right)}{2\mathrm{C}_s^t + 1}$$

ELSE

$$\mu_s^t = -\left(\frac{\mathrm{F}_s^t + \iota_{s+1}^t + \mu_s^{t+1} - \left(\mathrm{I}_s^t + \iota_s^{t+1} + \eta_s^{t+1}\right)}{2\mathrm{c}_s^t + 1}\right)$$

$$\eta_s^t = 0$$

END IF

$$\iota_s^t = \mathrm{I}_s^t + \iota_s^{t+1} + \eta_s^t + \eta_s^{t+1} - \left(\mu_s^t + \mu_s^{t+1}\right) \tag{12}$$

  END REPEAT
END REPEAT

The intuition behind expression (11) is that the sojourn time becomes maximal either when the flow is meager and/or when the average inventory is abundant. The first situation may occur if the flow cost F_s^t is high, ι_{s+1}^t is high, so next station is overloaded, and/or if μ_s^{t+1} is high, hence the sojourn time in the current station will be minimal in time period $t + 1$, meaning that there will be a lack of inventory in the next period, so it is not convenient to evacuate too much of it at time t. The inventory in the station should be abundant if its cost I_s^t is low, if ι_s^{t+1} is low so the station is unlikely to become overloaded in the next period and/or if η_s^{t+1} is low thus the sojourn time in the current station will be minimal in the next period $t + 1$. The explanation of expression (12) is that the dual variable ι_s^t grows, and therefore station s is overloaded, if the inventory cost I_s^t is high, if ι_s^{t+1} is high, hence the station will be overloaded on the next period and/or if η_s^t or η_s^{t+1} is high and therefore the station is working at its maximum sojourn time during periods t and $t+1$, so the inventory is high compared to the flow. Conversely, ι_s^t decreases if $\mu_s^t + \mu_s^{t+1}$ are high, hence the station is working at its minimum sojourn time during periods t or $t+1$ so the inventory is low compared to the flow.

Once the dual variables satisfying the necessary KKT condition are obtained, we calculate feasible, and therefore optimal primal variables f_s^{t*} and i_s^{t*}, from the first station and the first period to the last station S and last period T. Recalling that f_0^t and i_s^0 are non-negative parameters, the algorithm for obtaining the primal variables is:

Algorithm 2 (Primal Optimal Variables, Single-Product)

FOR $s = 1 \rightarrow S$ REPEAT

FOR $t = 1 \rightarrow T$ REPEAT

IF $\eta_s^t > 0$

$$f_s^{t*} = \frac{2i_s^{t-1} + f_{s-1}^t}{2\mathrm{C}_s^t + 1} \tag{13}$$

else

$$f_s^{t*} = \frac{2i_s^{t-1} + f_{s-1}^t}{2\mathrm{c}_s^t + 1} \tag{14}$$

END IF

$$i_s^{t*} = i_s^{t-1*} + f_{s-1}^{t*} - f_s^{t*} \tag{15}$$

END REPEAT

END REPEAT

Expressions (13), (14) and (15) are equivalent to expressions (2), (3) and (4) respectively. Algorithm 1 and Algorithm 2 run in $\mathrm{S} \cdot \mathrm{T}$ steps.

At this point we reconsider the non-negativity constraints for the flow and the inventory through the following proposition:

Lemma. If $\mathrm{c}_s^t \geq 0.5$ for $s = 1, \ldots, S$ and $t = 1, \ldots, T$, then Algorithms 1 and 2 always generate a solution with $\vec{f}^* \geq 0$ and $\vec{i}^* \geq 0$.

Proof. Consider the definition of flow obtained by expression (14), and replace it in expression (15), then:

$$i_s^{t*} = \frac{2\mathrm{c}_s^t \cdot f_{s-1}^{t*} + \left(2\mathrm{c}_s^t - 1\right) \cdot i_s^{t-1}*}{2\mathrm{c}_s^t + 1}.$$

Therefore, by construction, if $c_s^t \geq 0.5$ then the inventory i_s^{t*} will be non-negative with $i_s^0 \geq 0$, and from equations (14) the flows f_s^{t*} will also be non-negative. The case when the flow is obtained by expression (13) is similar, but the first case is more restrictive because $C_s^t > c_s^t$. □

Corollary. Algorithms 1 and 2 can find ${}_pf_s^t \geq 0$ and ${}_pi_s^t \geq 0$ that minimize (1) subject to (2), (3) and (4), with a pseudo-linear complexity.

Proof. If $\mathrm{c}_s^t \geq 0.5$ for all s and t, then the lemma above holds and the complexity is order of $S \times T$ by construction. If $c_s^t < 0.5$ for some particular s and t, the input data can be scaled by subdividing the time periods as much as necessary to meet the lemma's requirement. Suppose that $\underline{\mathrm{c}}_s^t$ is the smallest lower bound for the sojourn time. If time periods are divided, then their number increases by $\lceil 0.5/\underline{\mathrm{c}}_s^t \rceil$, making the complexity of the algorithm to be order of $S \times T \times \lceil 0.5/\underline{\mathrm{c}}_s^t \rceil$. □

3. Lagrangian relaxation algorithm

Since we have an efficient technique for solving the dynamic single-product multi-station system with no capacity constraints, we can dualize the family of constraints (5) related to the maximum throughput of each station. We include in the objective function the violation of the constraints weighted by the Lagrangian multipliers ${}^r\pi^t$ of every resource r at time period t. The multipliers must be non-negative as the constraints are of the "$\leq$" type. The problem to be solved is:

Minimize:

$$\sum_{p=1}^{P} ({}_p\mathbf{F}_p\vec{f} + {}_p\mathbf{I}_p\vec{i}) - \sum_{r=1}^{R}\sum_{t=1}^{T}\left({}^r\pi^t \cdot \left({}^r\mathrm{M}^t - \sum_{p=1}^{P}\sum_{s=1}^{S} {}_p^r\mathrm{U}_s^t \cdot {}_pf_s^t\right)\right) \tag{16}$$

Subject to:

$$\mathbf{A}^f{}_p\vec{f} + \mathbf{A}^i{}_p\vec{i} = 0 \quad \text{for } p = 1, \ldots, P \tag{17}$$

$${}_p\check{\mathbf{C}}^f\vec{f} + {}_p\check{\mathbf{C}}^i\vec{i} \leq 0 \quad \text{for } p = 1, \ldots, P \tag{18}$$

$${}_p\hat{\mathrm{C}}^f{}_p\vec{f} + {}_p\hat{\mathrm{C}}^i{}_p\vec{i} \leq 0 \quad \text{for } p = 1, \ldots, P \tag{19}$$

$${}^r\pi^t \geq 0 \quad \text{for } r = 1, \ldots, R; \quad t = 1, \ldots, T$$

The inventory constraints in expression (17) use the same matrices $\mathbf{A}^f$ and $\mathbf{A}^i$ for all the products, as their route through the different stations is assumed to be the same. The maximum and minimum sojourn time constraints (18) and (19) require different matrices, since such parameters are product-specific. Given the Corollary above, constraints ${}_p\vec{f} \geq 0$ and ${}_p\vec{i} \geq 0$ are ignored assuming ${}_p\mathrm{c}_s^t \geq 0.5$ for $p = 1, \ldots, P; s = 1, \ldots, S$ and $t = 1, \ldots, T$. To solve this new problem we propose an iterative technique, similar to the one discussed by Fisher (1979), where the variable cost multiplying ${}_p\vec{f}$ is modified according to how much the dualized constraints are violated. Defining ${}^r\pi_j^t$ as the value of ${}^r\pi^t$ at iteration j, expression (16) can be written as follows:

$$\begin{aligned}
&\sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} {}_p\mathrm{F}_s^t \cdot {}_p f_s^t + \sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} {}_p\mathrm{I}_s^t \cdot {}_p i_s^t + \sum_{r=1}^{R}\sum_{t=1}^{T}\sum_{p=1}^{P}\sum_{s=1}^{S} {}^r\pi_j^t \cdot {}_p^r\mathrm{U}_s^t \cdot {}_p f_s^t \\
&\quad - \sum_{r=1}^{R}\sum_{t=1}^{T} \left({}^r\pi_j^t \cdot {}^r\mathrm{M}^t\right) \\
&\quad = \sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} \left(\left({}_p\mathrm{F}_s^t + \sum_{r=1}^{R} {}^r\pi_j^t \cdot {}_p^r\mathrm{U}_s^t\right) \cdot {}_p f_s^t\right) + \sum_{p=1}^{P}\sum_{s=1}^{S}\sum_{t=1}^{T} {}_p\mathrm{I}_s^t \cdot {}_p i_s^t \\
&\quad - \sum_{r=1}^{R}\sum_{t=1}^{T} \left({}^r\pi_j^t \cdot {}^r\mathrm{M}^t\right)
\end{aligned} \tag{20}$$

From expression (20) we define ${}_p\mathrm{F}'^t_s = ({}_p\mathrm{F}_s^t + \sum_{r=1}^{R} {}^r\pi_j^t \cdot {}_p^r\mathrm{U}_s^t)$ as the new flow cost for Algorithm 3, which aims to find a close-to-optimal solution for the relaxed problem performing J iterations.

Algorithm 3 (Multi-Product)

DEFINE:
    ${}^r\pi_1^t = 0$ for $r = 1, \ldots, R$ and $t = 1, \ldots, T$.
    $\delta_1 = 0.0001$
FOR $j = 1 \rightarrow J$ REPEAT
    Define ${}_p\mathrm{F'}^t_s = ({}_p\mathrm{F}_s^t - \sum_{r=1}^{R} {}^r\pi_j^t \cdot {}_p^r\mathrm{U}_s^t)$ for $p = 1, \ldots, P; s = 1, \ldots, S$
        and $t = 1, \ldots, T$.
FOR $p = 1 \rightarrow P$ REPEAT
    Find dual variables for product p considering flow cost F'
    Find primal variables for product p
END REPEAT
${}^r\pi_{j+1}^t = {}^r\pi_j^t + \delta_j \cdot (\sum_{p=1}^{P}\sum_{s=1}^{S} {}_p^r\mathrm{U}_s^t \cdot {}_p f_s^t - {}^r\mathrm{M}^t)$
    $\delta_{j+1} \leftarrow$ heuristic (δ_j)
END REPEAT

The heuristic modifying δ makes $\delta_{j+1} = 10^{\delta j} - 1$ either when the objective function (20) does not change too much in successive iterations, or when the maximum flow constraints in expression (5) are violated. The heuristic makes $\delta_{j+1} = \log(\delta_j - 1)$ when neither the objective function nor the constraints change too much, or when either the objective function or the constraints oscillate too much around a given value. The parameters for defining "too much" are obtained by experimentation and then remain fixed for the computational tests.

4. Benchmark instances and computational results

The above algorithm is tested in instances with P products, S stations, R resources and T time periods, denoted as $P \times S \times R \times T$. Such instances have $2 \cdot P \cdot S \cdot T$ decision variables, $(4 \cdot P \cdot S + R) \cdot T$ inequality constraints, including non-negativity, and $P \cdot S \cdot T$ equality constraints. The parameters of the benchmark instances are randomly generated from uniform distributions U(lower limit, upper limit). There are $P \cdot T$ input flow parameters ${}_pf_0^t \sim \mathrm{U}(0, 10)$ and $P \cdot S$ initial inventory parameters ${}_pi_s^0 \sim \mathrm{U}(0, 10)$. There are $P \cdot S \cdot T$ flow cost parameters ${}_p\mathrm{F}_s^t \sim \mathrm{U}(0, 10)$, $P \cdot S \cdot T$ inventory cost parameters ${}_p\mathrm{I}_s^t \sim \mathrm{U}(0, 10)$, $P \cdot S \cdot T$ minimum sojourn time parameters ${}_p\mathrm{c}_s^t \sim \mathrm{U}(0.5, 10)$, $P \cdot S \cdot T$ maximum sojourn time parameters ${}_p\mathrm{C}_s^t \sim U({}_pc_s, {}_pc_s + 10)$ and $P \cdot S \cdot R \cdot T$ utilization parameters ${}_p^r\mathrm{U}_s^t \sim \mathrm{U}(0, 10)$.

There are $R \cdot T$ maximum availability of resource parameters ${}^r\mathrm{M}^t$ not randomly generated, otherwise the problem could become too constrained and therefore unfeasible, or alternatively too relaxed and hence trivial. As an alternative, ${}^r\mathrm{M}^t$ is defined as follows: The amount ${}^r\underline{\mathrm{M}}^t$ of resource r used by the system at time period t if its cost was zero and there was unlimited availability is obtained. For each resource $r\,{}^r\underline{\mathrm{M}}^{\max}$ is defined as the maximum ${}^r\underline{\mathrm{M}}^t$ over the T time periods, and then ${}^r\mathrm{M}^t = \alpha \cdot {}^r\underline{\mathrm{M}}^{\max}$ is defined, where α is the minimum value among the set $\{100\%, 95\%, 90\%, \ldots.\}$ such that the generated problem instance subject to the maximum flow constraint $\sum_{p=1}^{P} \sum_{s=1}^{S} {}_p^r\mathrm{U}_s^t \cdot {}_pf_s^t \leq \alpha \cdot {}^r\mathrm{M}^{\max}$ for $t = 1, \ldots, T$ is still feasible. In other words, for each resource we obtain its highest amount required in the planning time horizon, and then we lower it at a percentage α as many times as possible before it becomes infeasible. In summary, a $P \times S \times R \times T$ instance has $P \cdot (T + R) + P \cdot S \cdot T \cdot (4 + R) + R \cdot T$ parameters, where only $R \cdot T$ are calculated and the rest are randomly generated.

Algorithms 1 and 2 are specified in a straightforward manner in Microsoft Excel using standard formulas and cell references, while Algorithm 3 is implemented as a macro. Its performance is compared with the professional LP package Premium Solver Plus 3.5 from Frontline Systems, an enhanced version of the standard solver provided by Microsoft Excel. Although such software is by no means the industry standard for complex operation research problems, the ultimate purpose of this work is to provide the personnel of several offices of a public bureau described in Section 5 with a cost-effective planning tool. Although it is possible to link the Excel front-end with LP solvers such as CPLEX or IBM OSL, neither their speed nor their precision is required in the application where most parameters are approximations. Also, the implementation's complexity and the licenses would raise the cost of the solution unnecessarily.

Each row in Table 1 corresponds to a problem size tested with ten randomly generated instances solved in J iterations. The first four columns show the dimensions P, S, R and T. For each iteration number J, three values are displayed that compare the output of our algorithm to the optimal solution: The difference Δ O.F. between the objective functions calculated as ((optimal solution—algorithm solution)/optimal solution); the standard deviation σ O.F. of such difference for the ten instances; and the

Table 1
Results for instance with 10 products, stations and periods.

Instance size				$J=1$			$J=5$			$J=25$		
P	S	R	T	Δ O.F. (%)	σ O.F. (%)	Δ Cnst. (%)	Δ O.F. (%)	σ O.F. (%)	Δ Cnst. (%)	Δ O.F. (%)	σ O.F. (%)	Δ Cnst. (%)
10	10	0	10	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
10	10	1	10	7.4	3.0	66.4	1.9	1.1	7.1	1.2	0.7	4.2
10	10	2	10	5.3	2.6	41.1	1.2	1.0	4.0	1.1	0.9	3.3
10	10	3	10	5.0	2.7	37.7	1.4	0.5	4.3	1.4	0.6	3.6
10	10	4	10	6.0	2.9	42.2	2.0	0.8	4.2	1.9	0.7	3.6
10	10	5	10	8.9	4.2	55.8	2.3	0.8	4.3	2.3	1.0	3.9
10	10	6	10	9.1	5.0	53.4	2.7	0.7	5.2	2.2	0.6	3.9
10	10	7	10	8.0	3.9	37.9	2.3	1.0	4.1	2.3	0.7	4.0
10	10	8	10	11.6	4.4	57.8	3.1	1.1	4.6	2.7	0.9	3.4
10	10	9	10	8.1	2.7	50.9	2.3	0.6	4.0	2.0	0.3	3.3
10	10	10	10	9.1	2.8	50.1	3.3	1.4	4.9	3.1	1.0	4.3
Average				7.1	3.1	44.19	2.0	0.8	4.2	1.9	0.7	3.4

average violation Δ Cnst. of the maximum flow constraints. This last magnitude is a percentage of by how much our solution violates expression (5), because its throughput is above the system's capacity. Recalling that $(x)^+$ is the positive part of x,Δ Cnst. is de average of expression $(\sum_{p=1}^{P}\sum_{s=1}^{S} {}^{r}_{p}\mathrm{U}^{t}_{s} \cdot {}_{p}f^{t}_{s} - {}^{r}\mathrm{M}^{t})^{+}/\,{}^{r}\mathrm{M}^{t}$. As expected, when no resources are involved, the method is optimal in the first iteration. With one or more resources the solution is not primal feasible while the violation Δ Cnst. is significant. After five iterations, the value of the objective function provided by our algorithm converges to the optimum within a difference Δ O.F. of 2% on average, and Δ Cnst. falls to 4.2%. After 25 iterations Δ O.F. converges very slowly, while Δ Cnst. reaches an "asymptotic" 3.4%. Surprisingly, the deterioration of the performance of the method is small with respect to the number of resources R, regardless of the number of iterations.

Table 2 shows the results for several instances of different sizes, comparing the computational time of the LP package and our method, running on a Pentium personal

Table 2
Results for instance of different sizes.

P	S	R	T	No. of variables	No. of constraint	Solver time (secs)	Algorithm time (secs)	Δ O.F. (%)	Δ Cnst. (%)
10	6	15	6	360	1,890	27	16	1.8	4.3
10	8	15	8	640	3,320	76	22	2.1	4.3
10	10	15	10	1,000	5,150	185	30	1.8	4.4
10	12	15	12	1.440	7,380	514	41	1.6	4.4

computer, 166 MHz, 128 Mbytes of RAM, with Windows 95. The method's performance critically degrades in terms of numerical precision and computational times for larger instances. In summary, the method seems effective enough for small and middle-size instances, but with large instances the method suffers the same inconvenience the Premium Solver shows for middle-size ones. Therefore, we feel confident for applying it in a real-world problem as described in the next section.

5. Application to the planning of the Chilean General Treasury

The Chilean General Treasury is the governmental institution in charge of collecting debts related to taxes, credits, and fees. Its collection department employs around 300 people, including 41 lawyers, in 35 local offices across the country. The organization faces problems of inefficiency, partially caused by centralized decisions made at the headquarters in the capital city of Santiago that are oblivious of the local realities. Since the debt is worth more than 200 hundred million dollars a year for the state, any gain in productivity can be extremely profitable. In order to allow some autonomy and yet not lose control, Singer, Donoso, and Poblete (2002) formulate a linear program optimizing the working plan of a given local office for a time horizon of one year. This planning tool should be available for managers at the offices in spreadsheet format, due to the advantages explained by Leon, Przasnysky, and Seal (1997) and confirmed by Singer, Donoso, and Jara (2002) in planning the distribution of liquefied petroleum gas in Chile. As the parameters used are particular to each office, the tool allows for independent and objective operational planning.

The debt collecting process depicted by figure 3 starts when a person is charged (i) to have a debt related to income, real estate , sales, or other tax, and his/her data is registered in a system called United Tax Account (CUT). The headquarters decide when different types of debts should result in civil actions (ii) whereby the different local offices will prosecute the citizens until they pay their debts, or else their assets will be liquidated in a public auction. In the 1st *Stage* of the lawsuit, the collectors and support personnel notify the debtors and seize the property (iii). Once notified, the defendant can appeal, forcing the local office's lawyer to prepare the case (iv) and give the information to the treasurer in order for him to decide (v). At some point the file is sent to a civil court (vi) where a civil judge confirms or overrules the liquidation of the person's assets (vii). If he confirms, the 2nd *Stage* begins with a second notification (viii) so again the defendant can appeal and the lawyer must work in the case in order to litigate in front of the judge (ix). If there is no change in the decision of liquidation, the treasurer and the support personnel must perform a number of tasks (x) until the case starts the final stage of *Liquidation* (xi). In this stage the person's disposable belongings are confiscated and the information about the time, place and assets of the auction are advertised in a local newspaper (xii). The possessions are finally sold by an auction house (xiii). Certain actions may halt the process at any stage: payments of the debt, arrangements, clearings of database errors and *exclusions*, which correspond to decisions of terminating the prosecution due to lack of

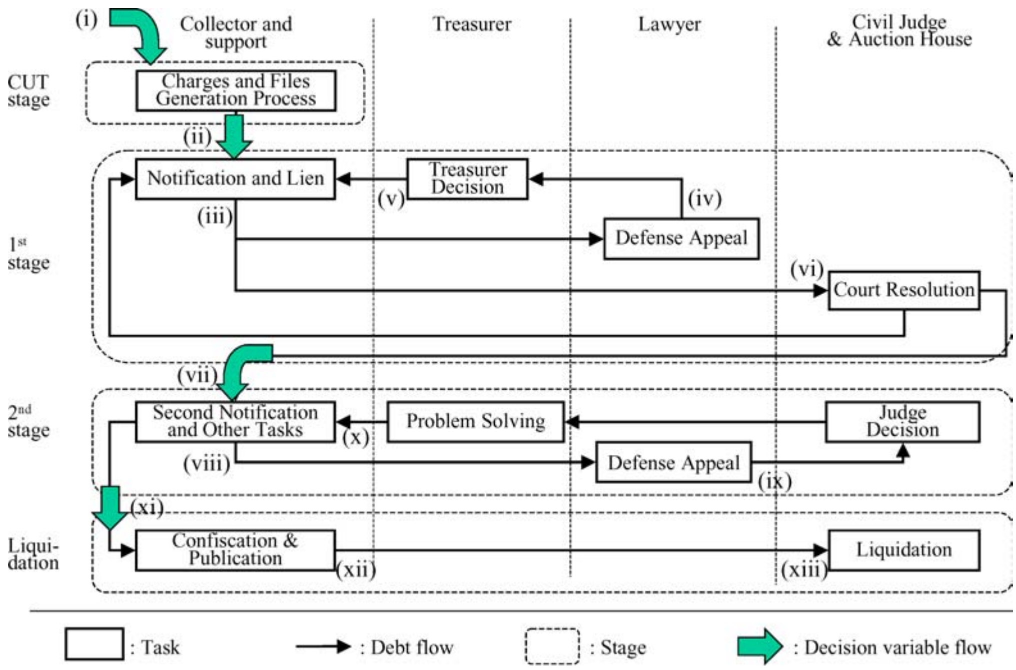

Figure 3. Flow diagram of the debt collecting process.

resources. There are maximum sojourn time constraints, since no civil action can remain in a given stage longer than a given time limit, or the case is lost due to legal prescription. There are minimum sojourn time constraints, since the courts require a minimum time to perform the legal procedures.

We model this process as a 4-station system because within each stage the civil actions may be subject to the same task several times, for instance notification, but once they cross to the following stage they do not return. Although the collecting process is similar for all debts, several parameters vary depending on the type and amount of the tax. We define eight categories for debts of similar characteristics such as value, sojourn time bounds and unit cost. Six types of resources are considered at each office: the treasurer, the lawyers, the collectors, the support personnel, the budget for publication and the budget for additional staff. The time horizon of one year is divided into six two-month periods. In summary, the problem at hand corresponds to an $8 \times 4 \times 6 \times 6$ instance, with $2 \cdot 8 \cdot 4 \cdot 6 = 384$ decision variables, so it is manageable by the algorithm described in Section 3.

In the prototype stage of the project of implementing a planning system for the Chilean General Treasury, four local offices at medium sized cities were considered: Iquique (1,800 km north of Santiago), Valparaíso (150 km west of Santiago), San Fernando (100 km south of Santiago) and Puerto Montt (1,000 km south of Santiago). In each office we conducted interviews during three days with the treasurer, lawyers, collectors

Table 3
Results for the four local offices of the Chilen General Treasury.

	$J = 5$		$J = 15$		$J = 50$	
Local office's city	Δ O.F. (%)	Δ Cnst. (%)	Δ O.F. (%)	Δ Cnst. (%)	Δ O.F. (%)	Δ Cnst. (%)
Iquique	5	20	1	13	2	13
Valparaiso	31	70	0	49	2	11
San Fernado	49	102	26	45	7	27
Puerto Montt	9	27	11	27	7	25
Average	24	55	9	33	4	19

and assistants, gathering documents and some statistics of past performance. In order to maintain objectivity, none of them were allowed to see the data provided by the other offices. Recalling that there are $8 \cdot 4 \cdot 6 = 192$ flow cost/benefit parameters of moving one civil action to the next stage, we only elicited $8 \cdot 4 = 32$ of the first time period, as the others were obtained by applying a discount rate. Inventory cost was not considered. The minimum and maximum sojourn time parameters are the same for all the time periods, so another $2 \cdot 8 \cdot 4 = 64$ parameters were elicited. There are $8 \cdot 4 \cdot 6 \cdot 6 = 1,152$ utilization parameter ${}^{r}_{p}\mathrm{U}^{t}_{a}$, but only $8 \cdot 4 \cdot 6 = 192$ of them were elicited as they are the same for all the time periods.

Table 3 presents the results of the planning algorithm applied on the four local offices of the Chilean General Treasury under study. The exclusion, i.e. the termination of the prosecution due to lack of resources, is an exogenous parameter obtained from Singer, Donoso, and Poblete (2002). These instances are heavily constrained by resource availability, as revealed by their corresponding shadow prices. As the dual variables related to the most restrictive constraints are estimated using the iterative method of Algorithm 3, instead of being exactly calculated by Algorithms 1 and 2, the method looses some of its effectiveness reported by Table 1. Notice that $8 \times 4 \times 6 \times 6$ instances are comparatively small with the ones presented in Table 2, so the computational times for both the Premium Solver and our method are below half a minute.

According to Lane (1998), besides computational efficiency, a model should also be examined in terms of its analytical quality of policy results and managerial insights. In this case the primal optimal variables represent the production goals for each time period in terms of the number of debts of each category to be transferred from one stage to the next. They can be obtained once a year, or alternatively every two moths following the rolling horizon technique described by Singer (2001) for the job shop, which schedules successive overlapping time-windows, fixing the most immediate part of them. The resulting objective function may be regarded as the collection goal for each office, depicted by the bars in figure 4. The figure also depicts the average Lagrangian multipliers for lawyers and for collectors, equal to the marginal benefit of increasing their

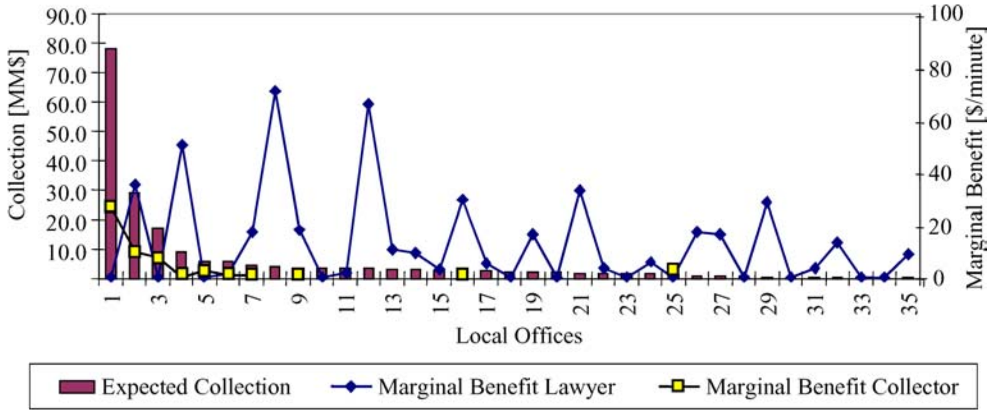

Figure 4. Collection goals and marginal benefits for lawyers and collectors.

availability by one additional minute. In general terms, there is a shortage of collectors only in the largest offices. In most of the smaller offices there are too many collectors, reason why the corresponding Lagrangian multipliers are zero, in which case they are not depicted in the graph. In contrast, there is an important shortage of lawyers in the smaller offices, especially in the ones labeled as 6, 8 and 12, where an additional attorney would be very productive.

6. Conclusions

The method presented aims to solve a linear problem using a technique traditionally used for non-linear formulations. In medium sized instances, a professional optimization package took several minutes to provide a solution in a spreadsheet, unacceptable for our purposes. Conversely, our algorithm implemented as a macro in Microsoft Excel reaches a solution in a few seconds. For $10 \times 10 \times 10 \times 10$ instances, after 25 iterations the objective function is in average 2% away from the actual minimum, with a constraint violation below 4%. If such a result is acceptable for the organization, our method not only provides rapid solutions, but also saves the license cost of linear programming packages. This is the case at the Chilean General Treasury, whose operations inspired the model under study. Since the elicited parameters are usually imprecise, the optimization is not supposed to generate mandatory instructions, but rather guidelines for the operation. Primal variables, obtained from input data specific for each local office, serve as guide for the next year's operations in terms of both activities and goals. Dual variables allow identifying bottlenecks and idle resources.

A side effect of our optimization method is a better understanding of the problem at hand. Expression (11) combining cost criteria, the inventory situation and the decisions made for successive stations and future time periods, provides a rule for deciding when a station should work as fast or slow as possible. Expression (12) formalizes the intuition about how overloaded a station is, or alternatively, how scarce are inventories. This may guide the production plan without having to implement the entire model, a matter that can be the subject of future research.

Acknowledgment

This research has been partially supported by FONDECYT project number 10270528.

References

Altiok, T. (2000). "Tandem Queues in Bulk Port Operations." *Annals of Operations Research* 93, 1–14.

Auslander, A., R. Cominetti, and M. Haddou. (1997). "Asymptotic Analysis for Penalty and Barrier Methods in Convex and Linear Programming." *Mathematics of Operational Research* 22, 43–62.

Berry, D., G. Evans, and M. Naim. (1998). "Pipeline Information Survey: A UK Perspective." *Omega* 26, 115–131.

Bradley, J. and B. Arntzen. (1999). "The Simultaneous Planning of Production, Capacity, and Inventory in Seasonal Demand Environments." *Operations Research* 47, 795–806.

Chen, F. (1998). "Stationary Policies in Multiechelon Inventory Systems with Deterministic Demand and Backlogging." *Operations Research* 46, S26–S34.

Dallery, Y. and S.B. Gershwin. (1992). "Manufacturing Flow Line Systems: A Review of Models and Analytical Results." *Queueing Systems Theory and Applications* 12, 3–94.

Fisher, M. (1979). "The Lagrangian Relaxation Method for Solving Integer Programming Problems." *Management Science* 22, 1–18.

FRONTLINE SYSTEMS 2000. Premium Solver Platform. www.frontsys.com

Ganesh, A.J. (1998). "Large Deviations of the Sojourn Time for Queues in Series." *Annals of Operations Research* 79, 3–26.

Govil, M. and M. FU. (1999). "Queueing Theory in Manufacturing: A Survey." *Journal of Manufacturing Systems* 18, 214–240.

Graves, S.C., D.B. Kletter, and W.B. Hetzel. (1998). "A Dynamic Model for Requirements Planning with Application to Supply Chain Optimization." *Operations Research* 46, S35–S49.

Lane, D.C. (1998). "Can We Have Confidence in Generic Structures?" *Journal of the Operational Research Society* 49, 936–947.

Leon, L., Z. Przasnysky, and K. Seal. (1997). "Spreadsheets and OR/MS Models: An End-User Perspective." *Interfaces* 26, 92–104.

Pinedo, M. and M. Singer. (1999). "A Shifting Bottleneck Heuristic for Minimizing the Total Weighted Tardiness in a Job Shop." *Naval Research and Logistics* 46, 1–17.

Rosling, K. (1998). "Optimal Inventory Policies for Assembly Systems Under Random Demands." *Operations Research* 37, 365–579.

Singer, M. (2000). "Forecasting Policies for Scheduling a Stochastic Due Date Job Shop." *International Journal of Production Research* 38, 3623–3637.

Singer, M. (2001). "Decomposition Methods for Large Job Shops." *Computers and Operations Research* 28, 193–207.

Singer, M., P. Donoso, and S. Jara. (2002). "Fleet Configuration Subject to Stochastic Demand: An Application in the Distribution of Liquefied Petroleum gas." *Journal of the Operations Research Society* 53, 961–971.

Singer, M., P. Donoso, and F. Poblete. (2002). "Semi-Autonomous Planning Using Linear Programming in the Chilean General Treasury." *European Journal of Operations Research* 140, 517–529.

Van Maaren, H. and T. Terlaky (1997). "Inverse Barriers and CES Functions in Linear Programming." *Operational Research Letters* 20, 15–20.

Tan, B. and S. Yeralan. (1997). "A Decomposition Model for Continuous Materials Flow Production Systems." *International Journal of Production Research* 35, 2759–2772.

Yang, J., H. Yan, and S. Sethi. (1999). "Optimal Production Planning in Pull Flow Lines with Multiple Products." *European Journal of Operational Research* 119, 582–604.