Discussion Paper No. 1231

Real-Time Hierarchical Resource Allocation

Timothy Van Zandt

April 7, 1997
Revised October 23, 1998

Math Center web site:
http://www.kellogg.nwu.edu/research/math

Real-Time Hierarchical Resource Allocation

Timothy Van Zandt*
Northwestern University and INSEAD

April 7, 1997
Revised October 23, 1998

Abstract

This paper presents a model that distinguishes between decentralized information processing and decentralized decision making in organizations; it shows that decentralized decision making can be advantageous due to computational delay, even in the absence of communication costs. The key feature of the model, which makes this result possible, is that decisions in a stochastic control problem are calculated in real time by boundedly rational members of an administration staff. The decision problem is to allocate resources in a changing environment. We consider a class of hierarchical procedures in which information about cost functions flows up and is aggregated by the hierarchy, while allocations flow down and are disaggregated by the hierarchy. Nodes of the hierarchy correspond not to a single person but to decision-making units within which there may be decentralized information processing. The lower tiers of multitier hierarchies can allocate resources quickly within small groups, while higher tiers are still able to exploit gains from trade between the groups (although on the basis of older information).

JEL Classifications: D83, D23
Keywords: decentralization, hierarchies, bounded rationality, real-time control

Author's address:

Math Center (CMS-EMS)	Voice:	+1 (847) 491-4414
2001 Sheridan Road, Room 371	Fax:	+1 (847) 491-2530
Northwestern University	Email:	tvz@nwu.edu
Evanston, IL 60208-2014	Web:	zandtwerk.kellogg.nwu.edu

---

*The comments of Roy Radner and Bob Vanderbei are greatly appreciated. This research was supported in part by grants SBR-9223917 and IRI97-11303 from the National Science Foundation, a CORE research fellowship (1993-1994), grant 26 of the "Pôle d'Attraction Interuniversitaire" program of the Belgian Government, and a grant from the University Committee on Research in the Humanities and Social Sciences at Princeton University.

Contents

1 Introduction **1**

2 Hierarchical decomposition of resource allocations **3**

3 Batch processing and delay **5**

3.1 What does the administrative staff do? 5

3.2 A model of decentralized batch processing 6

3.3 Discussion of the computation model 8

3.4 Decentralization and delay 9

4 Real-time decentralized information processing **11**

4.1 Measuring the cost of delay 11

4.2 A procedure with decentralized decision making 13

4.3 The benefits and costs of decentralized decision making 15

5 Interpretation **17**

6 Related literature on decentralization **18**

6.1 Information transmission costs 19

6.2 Incentives 19

6.3 Information processing 20

7 Further comments and extensions **21**

7.1 Optimality 21

7.2 Alternative procedures 21

7.3 Complications 23

Appendix: General hierarchies **24**

References **28**

1 Introduction

Consider the capital budgeting processes of large firms, or the procedures for allocating resources within large nonmarket organizations such as governments, firms, and universities. The flow of information in these procedures may resemble the hierarchical flow depicted in Figure 1. At the bottom of the hierarchy are the shops (or operatives, or whatever are the ultimate recipients of resources). In the upper tiers are managers or administrators, who are independent of the shops. Information about the shops is aggregated by a flow of information up the hierarchy, and resources are recursively disaggregated by a flow of information down the same hierarchy. These procedures exhibit both *decentralized information processing*—meaning that the resource allocations are calculated jointly by the members of the administrative staff—and *decentralized decision making*—meaning that (a) each node makes decisions that constrain the resource allocations and (b) the decisions of different nodes of the hierarchy are calculated using different information.

This paper models decision procedures with such hierarchical upward and downward flows of information. It distinguishes between and explains some of the advantages and disadvantages of the two forms of decentralization. The nodes of the hierarchies correspond to multiperson decision-making units (offices) within which there is decentralized information processing and aggregation of cost information. The disaggregation of resource allocations—i.e., the decentralized decision making—defines the hierarchical structure.

Our model is one of *real-time decentralized information processing*. This means that decisions in a stochastic temporal decision problem are calculated by boundedly rational members of an administrative staff. One component of this methodology is a distributed computation model—that is, a specification of how multiple agents jointly process information when the number of agents and the algorithms they follow are endogenous. The other component is a temporal decision problem—in our case, a resource allocation problem with a changing environment. Unlike the "ad hoc" approach to boundedly rationality—in which suboptimal decision rules are motivated informally by complexity—this paradigm models the process by which decisions are made, as well as the constraints and costs that this process introduces.

The advantage of *decentralized information processing* in our model—whether within or across nodes of the hierarchy—is that operations can be performed concurrently by several agents. This means, loosely, that delay is lower and resource are allocated based on more recent information, compared to when one person performs all the operations sequentially.

The advantage of *decentralized decision making* is that managerial nodes in the lower tiers can allocate resources within small groups using recent information, while nodes in higher tiers are still able to exploit gains from trade between the groups (although based on older information). Specifically, when resource allocations are disaggregated through the hierarchy, each office suballocates resources to its subordinates based only on the aggregate of these subordinates' cost information. This is beneficial because each office's information is more recent (since it is less aggregated) than that of its superior. (The main cost of this decentralization is an increase in the amount of computation.)

This advantage of decentralized decision making—although only demonstrated for a specific form of hierarchically decentralized decision making and for a specific decision problem—may be stated more broadly. Decision making is decentralized when there is a nexus of overlapping spheres of decision making; the hierarchically decentralized decision making depicted in Figure 1 is just one example. This paper illustrates how decentral-

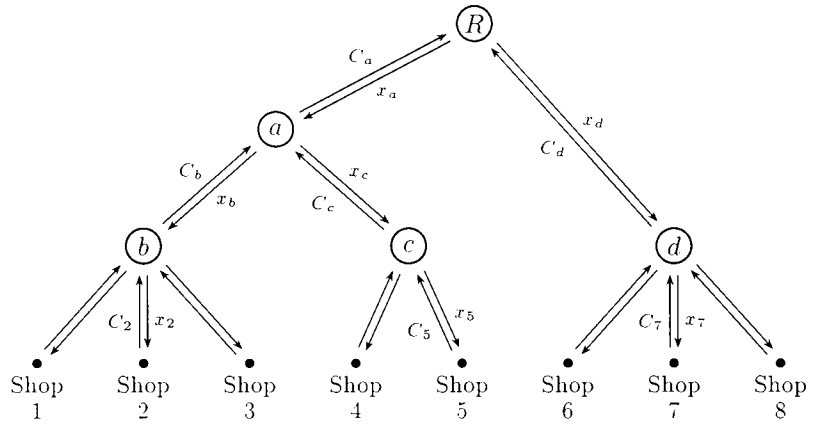

FIGURE 1. Hierarchical decomposition of the resource allocation problem without externalities. Cost functions are *aggregated* through an *upward* flow of information. Allocations are *disaggregated* through a *downward* flow of information.

ization allows decision making about small-scale coordination problems to use recent and hence better information—without precluding further coordination at a large scale based on older (but still valuable) information.

Our explanation for why (a) agents with no *prior* private information are hired to process information and (b) decision making is then decentralized among them—which is based on bounded rationality and delay—is meant to complement other explanations. Nevertheless, only a couple other papers endogenously derive both forms of decentralization (most notably, Geanakoplos and Milgrom (1991)). We refer the reader to Section 6 for a review of the literature; the next two paragraphs summarize a few main points.

The batch processing literature (e.g., Mount and Reiter (1990) and Radner (1993)—see Section 3 for more references) is also built on distributed computation and exhibits delay reduction due to decentralized processing. However, we show that a benchmark batch processing model would not explain the decentralized decision making seen in Figure 1. In batch processing, a given function is computed from data of the same lag, whereas the advantage of decentralized decision making in our model relies on the heterogeneity of the vintage of information across decision nodes.

Information transmission costs can easily explain decentralization of decision making to agents who are endowed with private information; this is the theme, for example, of the statistical theory of teams in Marschak and Radner (1972). However, hiring agents with no prior private information only aggravates these transmission costs. Incentive problems, which have dominated the economics literature on organizations, are hard pressed to explain either form of decentralization. On the other hand, such decentralization leads to many of the contracting and incentive problems that economists usually take as given, and hence understanding it is important for incentive theory.

That bounded rationality and delay are an important explanation for decentralized decision making is not a new idea. This theme arose several times in the debates in the 1930's and 1940's about socialism and economic institutions; for example, Hayek (1945, p. 524) states "we need decentralization because only thus can we ensure that the knowledge of the particular circumstances ... be promptly used". Whether these factors are more or less important than others is an empirical question, but we note that the ability to adapt to changing environments is often mentioned as a fundamental characteristic of

successful organizations. (See Van Zandt and Radner (1998) for further discussion.)

In this paper, we are able to derive the advantages and disadvantages of decentralized decision making—as well as to study methodological issues such as the comparison between batch processing and real-time processing, the relationship to the static decomposition of resource allocations, and the criteria for specifying a distributed computation model—in an abstract model that clarifies the results and demonstrates their generality. However, this abstraction does not permit the statistical assumptions required to characterize optimal hierarchies or perform comparative statics. In subsequent work, Van Zandt (1998c) studies a model that is similar but has very specific assumptions—the shops have quadratic cost functions whose parameters follow first-order autoregressive processes—which allow such calculations. That paper also provides a formal treatment of decision procedures and hierarchical structures and an analysis of the distributed statistical inference problem. Van Zandt (1998e) then characterizes the shape of optimal hierarchies, returns to scale, and comparative statics. Interestingly, firm size is limited in the model even though internal decentralization is possible. That paper also shows, for example, that organizations tend to be smaller and more interally decentralized the more quickly the environment is changing. Van Zandt (1998c, 1998e) contain additional references and a discussion of related team theory models—especially Geanakoplos and Milgrom (1991)—that are based on a static version of the decision problem.

Van Zandt (1998c, 1998e) further demonstrate the power of the model developed in this paper. Although there are other ways to obtain either decentralized information processing or decentralization decision making, this one paints a rich, dynamic picture of large administrative apparatuses, yet yields (in Van Zandt (1998c)) a tractable model with a specific characterization of the distribution of information in the organization and of the properties of optimal hierarchies.

Reader's guide We begin, in Section 2, by reviewing the notion that it is at least possible to hierarchically decompose the allocation of resources when there are no externalities—without explaining why it is advantageous to do so. In Section 3 we exclude various potential explanations not based on bounded rationality. Then we attempt, but also fail, to answer the question using a benchmark model of decentralized batch processing. The real-time model is presented in Section 4, where we compare two classes of decision procedures; one is identified with two-tier centralized hierarchies and the other with three-tier decentralized hierarchies. (General hierarchies are defined in the Appendix.) An interpretation is given in Section 5. Section 6 reviews other explanations of decentralization, and Section 7 discusses several extensions to and questions about the model (as well as additional related literature). For example, it explains that the underlying model is rich enough to construct decision procedures that resemble market mechanisms such as bilateral trade or wholesale/retail networks, thereby addressing other questions about market and nonmarket institutions and comparing their computational efficiency.

2 Hierarchical decomposition of resource allocations

Consider the following one-good resource allocation problem without externalities, framed as a cost minimization problem of an organization such as a firm. Fix a domain $X = \mathbb{R}$ or $X = \mathbb{R}_{++}$ for allocations. Given a total quantity $x_R \in X$ of a resource, the firm chooses an allocation $\{x_1, \ldots, x_n\}$ of the resource to n production shops or operatives in

order to solve

$$\text{(MAX)} \qquad \min_{\{x_i \in X\}_{i=1}^n} \sum_{i=1}^n C_i(x_i) \quad \text{s.t.} \quad \sum_{i=1}^n x_i = x_R \ .$$

The data in the problem are the shops' cost functions $\{C_1, \ldots, C_n\}$ and the amount x_R of the resource. The system is closed in the sense that x_R is not an endogenous allocation to the whole organization.

This is a canonical resource allocation problem with many interpretations. The variable x_i is a transfer to shop i that could represent capital, some other input, or orders to be filled. The function $C_i : X \to \mathbb{R}$ could represent a cost or negative profit, and it could be a reduced form that subsumes further unmodeled decisions taken within each shop. For example, $C_i(x_i)$ could be shop i's minimum cost of producing a quantity x_i of output; $-C_i(x_i)$ could be shop i's maximum profit when it has capital x_i. The resource could also be a consumption good and each "shop" i could be a consumer whose weighted utility in an aggregate welfare function is function $-C_i$. If $x_R = 0$, then the allocations represent net trades. Because costs are additively separable across shops, there are no externalities.

In the rest of this section, we review why it is at least possible, if not desirable, to solve this problem by hierarchically decomposing the allocation of resources (as shown in Figure 1). This is an example of the kind of decompositions of decision problems that are studied in Bernussou and Titli (1982) and Dirickx and Jennergren (1979).

We begin by defining the aggregate of a set of cost functions. Let $\bar{\mathbb{R}} = \mathbb{R} \cup \{-\infty\}$ and let $\bar{\mathcal{C}}$ be the set of functions from X into $\bar{\mathbb{R}}$, the elements of which we call *cost functions*. The *aggregate* (or "infimal convolution", as defined in Rockafellar (1970)) of a finite collection $\{C_k \in \bar{\mathcal{C}}\}_{k \in K}$ of cost functions is the cost function $C : X \to \bar{\mathbb{R}}$ defined by

$$C(x) \equiv \inf_{\{x_k \in X\}_{k \in K}} \sum_{k \in K} C_k(x_k) \quad \text{s.t.} \quad \sum_{k \in K} x_k = x$$

for $x \in X$; this is denoted by $\bigoplus_{k \in K} C_k$. If $C_A, C_B \in \bar{\mathcal{C}}$, then $\bigoplus_{k \in \{A,B\}} C_k$ is also denoted $C_A \oplus C_B$. The binary operation $\oplus : \bar{\mathcal{C}} \times \bar{\mathcal{C}} \to \bar{\mathcal{C}}$ is associative and commutative, and if $C_k \in \bar{\mathcal{C}}$ for $k = 1, \ldots, m$ then $\bigoplus_{k=1}^m C_k = C_1 \oplus \cdots \oplus C_m$.[1]

Now consider a hierarchy (i.e., a tree) like the one in Figure 1. The leaves are the n shops $\{1, \ldots, n\}$. Let J be the set of interior nodes, which we refer to as offices or managerial nodes. The root node in J is denoted by R and is also called the "center". For each office $j \in J$, let Θ_j be the set of j's immediate subordinates, which may contain both offices and shops, and let θ_j be the set of shops that are inferior to office j in the hierarchy; θ_j is called office j's *division* or simply "division j". In Figure 1, $\Theta_a = \{b, c\}$ and $\theta_a = \{1, 2, 3, 4, 5\}$. A node's tier is the maximum number of edges from the node to one of the leaves inferior to the node. Each of the leaves is in tier 0, and the *root* is the unique node in the highest tier, which we call the *height* of the hierarchy. (This notation is listed in Table 4 in the Appendix.)

Office j's aggregate cost function is $C_j \equiv \bigoplus_{j \in \theta_j} C_i$. That is, for each $x_j \in X$, $C_j(x_j)$ is the infimum of the total costs of the shops in θ_j when quantity x_j is allocated to these shops. It is the lack of externalities that makes C_j independent of the resources allocated

[1] We could define the aggregate of cost functions $\{C_k : X_k \to \bar{\mathbb{R}}\}_{k \in K}$ with different domains, in which case the domain of the aggregate cost function is $\sum_{k \in K} X_k$. To simplify notation, we selected a common domain X such that $X = \sum_{k \in K} X$, and hence aggregate cost functions also share the domain X. We also chose X to be open so that, in later sections, we can characterize solutions to (MAX) by first-order conditions.

to shops that are not in θ_j. Because the aggregation of cost functions can be decomposed into the binary associative and commutative operation $\oplus$,

$$C_j \;=\; \bigoplus_{i\in\theta_j} C_i \;=\; \bigoplus_{k\in\Theta_j}\left(\bigoplus_{i\in\theta_k} C_i\right) \;=\; \bigoplus_{k\in\Theta_j} C_k \;.$$

In Figure 1 this means, for example, that office a can calculate its aggregate cost function either by directly aggregating the cost functions $C_1,\ldots,C_5$ of shops 1–5 or by aggregating the aggregate cost functions C_b and C_c of offices b and c.

Furthermore, we can decompose not only the aggregation of cost functions, but the disaggregation of resource allocations, as follows. Let $\{x_i^* \in X\}_{i=1}^n$ be such that $\sum_{i=1}^n x_i^* = x_R$. For $j \in J$, let $x_j^* = \sum_{i\in\theta_j} x_i^*$. Then $\{x_i^* \in X\}_{i=1}^n$ solves (MAX) if and only if, for each $j \in J$, $\{x_k^*\}_{k\in\Theta_j}$ solves

$$\min_{\{x_k\in\mathbb{R}\}_{k\in\Theta_j}} \sum_{k\in\Theta_j} C_k(x_k) \quad \text{s.t.} \quad \sum_{k\in\Theta_j} x_k = x_j^* \;.$$

For example, in Figure 1, if office a has to allocate an amount x_a of the resource to the shops in division a in order to minimize the division's total costs, it can either do so directly or instead (a) allocate amounts x_b and x_c of the resource to offices b and c, in order to minimize the sum $C_b(x_b) + C_c(x_c)$ of their aggregate costs, and then (b) instruct each office b and c to divide its allocation among the shops in its division, in order to minimize the sum of those shops' costs.

This suggests the following hierarchical procedure. Recursively starting at the bottom of the hierarchy, each office calculates its own aggregate cost function from those of its subordinates, and sends the result to its superior. This stage ends when the root has calculated the overall aggregate cost function. Then, recursively starting with the root, each office allocates to its subordinates resources received from its superior, in order to minimize the total (aggregate) costs of its subordinates. The upward flow of cost functions and downward flow of resource allocations is then as shown in Figure 1.

3 Batch processing and delay

3.1 What does the administrative staff do?

Section 2 reviews the *possibility* of decomposing a resource allocation problem without externalities, as depicted in Figure 1. However, it does not explain why this would be advantageous.

In order to explain why the administrators in an organization would use a hierarchical procedure to allocate resources, we must explain why there are so many administrators in the first place. The reason, of course, is that the task of aggregating information and making decisions is too large for a single administrator. However, this obvious answer is not provided by the behavioral model of full rationality that is the foundation of most economic theory, *even when there are information transmission costs and incentive problems*. A single, fully rational CEO could instantly aggregate the relevant information about the shops and decide how to allocate resources to them. Transmission costs could lead such an entrepreneur to decentralize some processing tasks to the shops—if the latter are endowed with private information about their costs—but not to delegate such tasks to administrators who have no private information when hired. Incentive problems may lead the entrepreneur to hire administrators whose sole job is to audit or watch subordinates so that they do not shirk or lie, as in Calvo and Wellisz (1980) and Qian (1994).

But in this case other processing tasks would not be delegated to these agents, since such delegation would just create problems of private information that did not exist before. Instead, all agents (including auditors or supervisors) should communicate directly with the entrepreneur through a direct revelation mechanism. (See Section 6 for further discussion and exceptions.)

Thus, to explain the existence and activities of the administrative apparatus in Figure 1, we should model the bounded processing capacities of the individual administrators.

3.2 A model of decentralized batch processing

The nature of information processing constraints is that people are bounded in the amount they can process in a given amount of time. It is possible to suppress the temporal aspect of these constraints and simply bound the total amount of processing an agent can do, as in Geanakoplos and Milgrom (1991) and in Marschak and Reichelstein (1995, 1998). However, like the current paper, most of the economics literature on processing information with an endogenous number of agents has instead emphasized this temporal aspect.

One approach that emphasizes computational delay is decentralized batch processing, known in computer science as parallel or distributed batch processing (see Zomaya (1996)). Kenneth Mount and Stanley Reiter, starting in 1982, have advocated this as a model of human organizations. Models of organizations based on decentralized batch processing include Beggs (1995), Bolton and Dewatripont (1994), Friedman and Oren (1995), Malone and Smith (1988), Meagher and Van Zandt (1998), Mount and Reiter (1990, 1996), Orbay (1997), Radner (1993), Reiter (1996), and Van Zandt (1998d). The value of decentralizing information processing in those papers is typically that it reduces delay; in the periodic models of Bolton and Dewatripont (1994), Radner (1993), and Van Zandt (1998d), it also increases the rate (throughput) at which problems can be computed.

The first part of a decentralized batch processing model is a function $f\colon Y \to Z$ to be computed. The input domain Y is typically multidimensional, in which case all the data in the vector $y \in Y$ are available when the computation starts. If the output space Z can be written as a product, then delay is measured by the interval between when the computation starts and when all the components of $f(y)$ are calculated, even if some components are available at earlier times.

The other part of a decentralized batch processing model is a decentralized computation model, which consists of the following components:

1. a set of elementary operations—these are functions that, when composed, can yield f;
2. a description of how the processing activities of agents are coordinated and how information is communicated between agents—this may include a communication protocol;
3. a set of potential information processing agents, each of whom is characterized primarily by the time it takes the agent to perform each elementary operation and each operation in the communication protocol.

Given a decentralized batch processing model, a *procedure* (algorithm) specifies how one or more agents calculate $f\colon Y \to Z$ by performing elementary operations and sharing information.

For the resource allocation problem (MAX) in Section 2, the function to be computed is $f : \mathcal{C}^n \times X \to X^n$, where $f(C_1, \ldots, C_n, x_R)$ is the solution to (MAX) and $\mathcal{C} \subset \bar{\mathcal{C}}$ is the set of potential cost functions. Assume that $\mathcal{C}$ is a set of strictly convex and differentiable cost functions such that $\mathcal{C}$ is closed under the operation $\dot{+}$ and such that the resource allocation problem (MAX) has a solution for all $\{C_i \in \mathcal{C}\}_{i=1}^n$ and $x_R \in X$.

As will be explained in Section 3.3, we have some discretion in choosing the set of elementary operations. The following suit the objectives of this paper:

1. (aggregation of two cost functions) $f_1 : \mathcal{C}^2 \to \mathcal{C}$, where $f_1(C_A, C_B) = C_A \dot{+} C_B$;
2. (derivative of a cost function) $f_2 : \mathcal{C} \times X \to \mathbb{R}$, where $f_2(C, x) = C'(x)$;
3. (inverse derivative of a cost function) $f_3 : \mathcal{C} \times X \to \mathbb{R}$, where $f_3(C, p) = C'^{-1}(p)$.

These elementary operations are sufficient to compute $f(C_1, \ldots, C_n, x_R)$ as follows.

1. The aggregate cost function $C_R := C_1 \dot{+} \cdots \dot{+} C_n$ can be calculated with $n - 1$ of the operation f_1.
2. The shadow price $p_R := f_2(C_R, x_R)$ of the resource can be calculated with one operation f_2.
3. The allocation of each shop i can then be computed by setting the shop's marginal cost equal to the shadow price: $x_i := f_3(C_i, p_R)$. A total of n of the operation f_3 are needed.

Table 1 in Section 3.4 summarizes the elementary operations and the number that are performed in order to calculate $f(C_1, \ldots, C_n, x_R)$.

Next we specify the means by which agents communicate and are coordinated. We choose the simplest specification:

1. agents are coordinated by synchronously executing instructions (the organization's managerial procedures;
2. there are neither individual nor network communication costs or delays.

Finally, we specify the capacities of the potential information processing agents. We have already assumed that they are unconstrained in the communication abilities. We also assume the following:

1. agents have identical computational abilities and wages;
2. each elementary operation takes the same amount of time, which we call a *cycle*;
3. each agent is paid only when performing an operation, at a fixed wage per operation;
4. each agent has unbounded memory.

Overall, we have the simplest possible specification of the communication between and coordination of agents. The lack of communication costs and delays means that agents have equal access to the information in each other's memory, so we can imagine that there is actually a single memory shared by all the agents, and that each agent is a control unit or processing element that performs operations on the shared memory. It is arbitrary

which agents perform which operations each cycle—as long as no agent performs more than one operation at a time. Such a model is called a parallel random access machine (PRAM) in computer science.[2]

A computation procedure for this simple model can be specified by the operations to be performed each cycle, with each operation's inputs identified as available data or as the output of a previously performed operation. As in this paper, it is typically possible to describe computation procedures informally yet clearly. However, Van Zandt (1998c) defines the computation model and computation procedures more formally.

3.3 Discussion of the computation model

We have chosen a very simple model in terms of both the decomposition of the problem into elementary operations and the interaction between information processing agents. In this section, we explain the reasons for doing so. (The reader can opt to skip this discussion of methodology and proceed directly to Section 3.4.)

Consider first the selection of elementary operations. One of the reasons for decomposing the computation problem into elementary operations, whether processing is serial or in parallel, is to compare the delay and processing costs of different types and sizes of problems. To do so, we must decompose the different problems into a common set of operations. In Van Zandt (1998e), for example, we compare the information processing for different numbers n of shops. Therefore, the aggregation $C_1 \oplus \cdots \oplus C_n$ of all the cost functions should not be a single elementary operation, since this operation is different for different values of n. This is why the elementary operations we defined are all independent of n.

However, there are many sets of elementary operations that satisfy this invariance condition and that are sufficient to calculate $f : \mathcal{C}^n \times X \to X^n$. We must balance other goals when choosing among the possible specifications. On the one hand, a coarse decomposition is simpler. On the other hand, a fine decomposition permits more decentralized processing (the assignment of different elementary operations to different agents) and a more complete and realistic description of the actual activities of the agents. Because the purpose of this paper is to illustrate decentralization as simply as possible, we have opted for a very coarse (and unrealistic) set of elementary operations,[3] but the decomposition is still fine enough to permit decentralized processing.

The other simple features of this computation model suppress some potentially interesting issues, such as the problem of economizing memory and communication costs, the coordination of agents who are not synchronized, the scheduling of tasks when agents must be paid even when idle, and the assignment of tasks to agents with heterogeneous computation abilities and wages. However, these issues are orthogonal to and would obscure the main theme of this paper, which is that decentralized decision making can arise owing to computational delay, even in the absence of other processing constraints or heterogeneity among information processing agents.

---

[2]See Zomaya (1996) for an overview of the PRAM and other models of parallel computation. Our model is also a special case of the one in Mount and Reiter (1990). Their formalization deals with certain technical issues related to real-number computation and to the measurement of communication costs, which our model suppresses.

[3]Indeed, the operation of aggregating two cost functions is generally not "elementary" at all; see the Appendix for an example of iterative adjustment procedures that avoid the aggregation of entire functions.

Calculation	Elementary operation	Number of operations	Parallel delay
$C_R := C_1 \oplus \cdots \oplus C_n$	$f_1(C_A, C_B) = C_A \oplus C_B$	$n-1$	$\lceil \log_2 n \rceil$
$p_R := C'_R(x_R)$	$f_2(C, x) = C'(x)$	1	1
$\{x_i := C_i'^{-1}(p_R)\}_{i=1}^n$	$f_3(C, p) = C'^{-1}(p)$	n	1
	Total:	$2n$	$2 + \lceil \log_2 n \rceil$

TABLE 1. Elementary operations and serial and parallel delay for the resource allocation problem.

Note, however, that the computational delay upon which this theme relies could be due either (a) to human delays in reading, understanding and interpreting information or (b) to human delays in calculating with information they have already "loaded" into their brains. This is explained, although in the context of a different model, in Van Zandt (1998b, Section 5). We have chosen a model that contains delays only of type (b) both for simplicity and because delays of type (a) would introduce costs of communication between agents (the managerial wages for the time it takes agents to read messages sent by other agents). The reader might then have more trouble seeing that the decentralized decision making that arises in the model is due to delay rather than to communication costs.

The lack of communication costs in our model means that we are not attempting to examine two themes that have been important in most of the economics literature on decentralized batch processing.

1. One is that decentralizing *information processing* entails a trade-off between delay and communication costs. As more administrators share the operations and hence more operations are performed concurrently, delay is decreased but communication costs increase. Our computation model captures only one side of this trade-off, the reduction in delay. However, this paper is concerned not with that trade-off but instead with the trade-offs entailed by decentralizing *decision making*.

2. The other is the pattern of communication between individual administrators, which has been interpreted as an organization's structure. Without communication costs, this microcommunication is indeterminate; when all agents have equal access to all information at all times, the identities of the agents who perform the operations each cycle are not relevant to the performance of the procedure. However, this paper, as well as Van Zandt (1998c, 1998e), contend that organizational structure should be derived from the macro structure of decision making more than from the micro structure of message exchange between individual agents.

3.4 Decentralization and delay

This section describes two batch processing procedures and illustrates how decentralization reduces delay.

In the first procedure, a single agent (the "entrepreneur") calculates the resource allocation. The $2n$ operations listed in Table 1 must be performed sequentially by this agent, so the delay is $2n$.

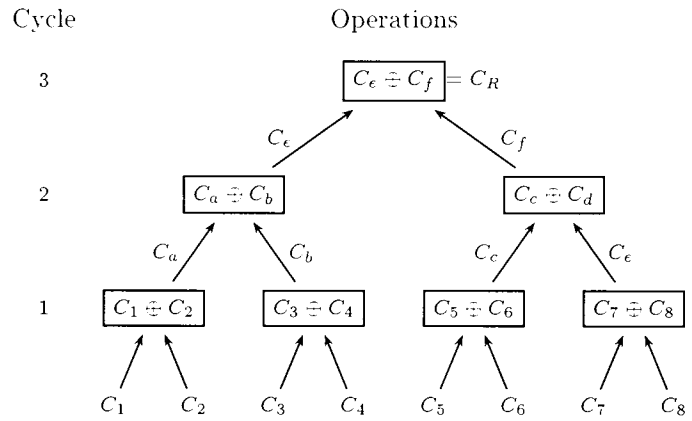

FIGURE 2. Associative computation by a PRAM.

Compare this with *decentralized* information processing, in which potentially many agents compute the resource allocation jointly. In the first stage, $C_R := C_1 \oplus \cdots \oplus C_n$ is calculated. The efficient algorithms for associative computation with a PRAM are illustrated in Figure 2. In the first cycle, the cost functions are divided into pairs, and each pair is assigned to a different agent, who calculates the aggregate of the two cost functions. In each subsequent cycle, the aggregate cost functions computed in the previous cycle are grouped into pairs, and again each pair is assigned to an agent who calculates the aggregate. The number of partial results is reduced by half in each cycle, so there is a single pair in cycle $\lceil \log_2 n \rceil$ whose aggregate is C_R. (The brackets $\lceil \cdot \rceil$ denote the ceiling or round-up operation.) Hence, the delay is $\lceil \log_2 n \rceil$ rather than the $n-1$ cycles it takes a single agent to compute C_R. However, the computation of C_R still requires $n-1$ operations.

The next step is to compute $p_R := f_2(C_R, x_R)$, which one agent does in one cycle. Finally, the n operations $\{x_i := f_3(C_i, p_R)\}_{i=1}^n$ can be assigned to n different agents and executed concurrently in one cycle. As summarized in Table 1, the total delay when the computation is decentralized is $2 + \lceil \log_2 n \rceil$, compared to $2n$ when a single agent calculates the allocations. *This reduction in the delay is the benefit of decentralization.*

The only administrative (computation) costs in this model are the wages of the agents. These are proportional to the total number of operations, which is $2n$ whether the computation is performed by one agent or many. Hence, there is no administrative overhead incurred by decentralization. This is because we assume that there are no communication costs and that agents are paid only for the operations they perform. Under different assumptions, such as in Radner (1993), increasing the number of agents who jointly calculate f reduces the delay but increases the administrative costs.

How does this batch processing model relate to Figure 1? On the one hand, we will see that the aggregation of cost functions in Figure 1 is similar to the decentralized aggregation of cost functions in our batch processing model. However, the disaggregation of resource allocations in Figure 1 has no analog in the batch processing model.

Consider first the aggregation of cost functions. If we treat each interior node in Figure 1 as an information processing agent in our computation model, then it takes an agent $j \in J$ who has s_j immediate subordinates a total of $s_j - 1$ cycles to compute her aggregate cost function. Agents b, c, and d can start this calculation at the same time; they finish after two, one, and two cycles, respectively. Then agent a computes $C_a := C_b \oplus C_c$ in

the third cycle and agent R computes $C_R := C_a \oplus C_d$ in the fourth cycle. The total delay of 4 is less than the delay of 7 for a single agent. This is similar to the parallel computation of C_R that is shown in Figure 2. (Although the nodes in Figure 2 are operations, it is possible to assign each operation to a different agent, in which case the nodes in Figure 2 correspond to agents.)

In contrast, *there is no analog in our batch processing model to the hierarchical disaggregation of resource allocations.* In the computation procedure described in this section and shown in Table 1, resources are allocated to all the shops in a *single step*,once the shadow price p_R is calculated. Suppose instead that, as in Figure 1, each agent j receives the allocation x_j for division j, calculates division j's shadow price $p_j := C'_j(x_j)$, and then allocates resources to each subordinate $k \in \Theta_j$ by setting x_k so that $C'_k(x_k) = p_j$. Then the shadow price at every node is the same. The only useful operations are the calculation of the overall shadow price, $p_R := C'_R(x_R)$, and of the individual shops' allocations, $x_i := C'^{-1}_i(p_R)$. The calculations of intermediate shadow prices and allocations increases not only the number of operations (by twice the number of intermediate nodes) but also the delay (by twice the number of intermediate tiers).

4 Real-time decentralized information processing

4.1 Measuring the cost of delay

For the moment, ignore the question of why resource allocations might be disaggregated hierarchically. Instead, motivate the next step by supposing that we have a batch processing model with communication costs and that we have derived the set of efficient procedures, within which there is a trade-off between delay and administrative costs (as is done, for example, in Radner (1993) and Van Zandt (1998d)). In order to determine which procedures are optimal or to study how information processing constraints affect returns to scale, we need to measure the administrative costs and the "cost" of delay. Administrative costs are easy to measure—for example, by managerial wages. Delay, on the other hand, is not an input that we can buy at a constant unit price. Instead, it has a decision-theoretic cost—higher delay means that decisions are based on older information. To quantify the costs of delay, we need a temporal decision problem in which current decisions are computed from lagged information. A decision procedure is then a decision rule together with a computation procedure for computing the decision rule. The computation of the decision rule must adapt to the timing of the arrival of information and of the decision epochs. This is a problem of real-time or on-line control.

We obtain a simple temporal version of the resource allocation problem in (MAX) by assuming there are discrete time periods $t \in \{\dots, -1, 0, 1, \dots\}$ and that, at the beginning of each period t, new cost functions $\{C_{1t}, \dots, C_{nt}\}$ are realized and observed and a deterministic quantity x_{Rt} of the resource must be allocated. That is, the shop costs in period t when the allocation of x_{Rt} is $\{x_{1t}, \dots, x_{nt}\}$ are $\sum_{i=1}^n C_{it}(x_{it})$. We are assuming that the resource constraint must be satisfied each period, and hence there is no intertemporal allocation of resources. However, the informational structure is dynamic because allocations are computed from past observations of the cost functions.[4]

We can use the same computation model as in Section 3 for the computation of the decision rules, but we need to specify the relationship between a cycle (the unit of time

[4]The reason we must assume that $\{x_{Rt}\}_{t=-\infty}^{\infty}$ is a deterministic process is that otherwise, since allocations must be calculated from past data, it could be impossible to calculate a feasible allocation.

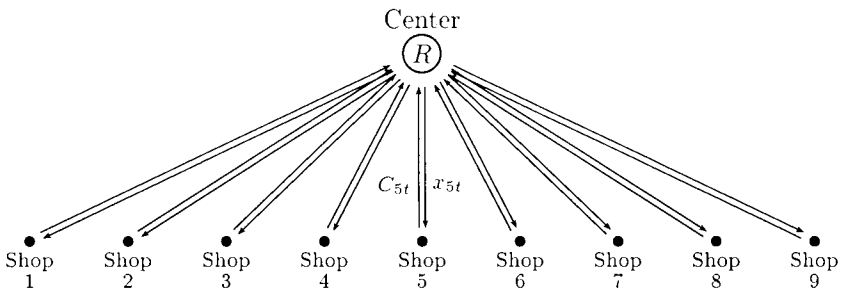

FIGURE 3. A two-tier centralized hierarchy.

in the computation model) and a period (the unit of time in the decision problem). We assume that a cycle and a period are the same; this assumption simplifies notation but is not important for the qualitative results.[5]

We measure the cost in each period by the sum of the administrative costs (of the computation procedure) and the expected shop costs (of the decision rule). In order to determine the expected shop costs for a decision rule, we need assumptions about the stochastic process $\{C_{1t}, \ldots, C_{nt}\}_{t=-\infty}^{\infty}$ (as in Van Zandt (1998c, 1998e)). However, for the purpose of this paper—which is to derive a qualitative rather than quantitative value of decentralized decision making—the reader should simply imagine that we have imposed statistical assumptions such that decision rules that use old information "tend" to have higher expected shop costs than those that use recent information.

In this real-time setting, the decision procedures that most closely resemble batch processing are stationary and compute the allocation for each period from data of homogeneous lags. Consider the following example. The resource allocation for each period t is calculated by collecting the cost functions $\{C_{1,t-d}, \ldots, C_{n,t-d}\}$ in period $t-d$ and then calculating the resource allocation using these cost functions and x_{Rt} as data, in the manner described in Section 3.[6] The lag d is the delay in performing these computations, which is given in Table 1. Hence, with serial processing (one agent), the allocation in each period is calculated from the cost functions from $2n$ periods ago, whereas with decentralized computation, each allocation is calculated from the cost functions from $2 + \lceil \log_2 n \rceil$ periods ago. Greater decentralization leads to lower delay—and hence better decision rules and lower shop costs—but might increase administrative costs (in a different computation model).

We consider this decision procedure to be a two-tier hierarchy, as shown for $n = 9$ in Figure 3. The single administrative node, which is the root node or center, is an office within which reside the agents who compute the decision rule. Each period, this office collects the current cost functions from the shops and sends the current allocation to the shops. In period t, the organization is busy computing the allocations for periods

[5]A possible justification is that, if the decision problem is actually in continuous time, then a natural interval for the purpose of collecting data and updating allocations is the discrete time unit in the computation model.

[6]The allocation in period t would thus be optimal if the cost functions were the same as in period $t-d$. With specific statistical assumptions, as in Van Zandt (1998c), we could allow the decision rule to take into account the expected changes in the cost functions between periods $t-d$ and period t. Note that, since the shadow price is calculated from $\{C_{i,t-d_i}\}_{i=1}^{n}$, the final step in which marginal cost is set equal to the shadow price must use these same cost functions (rather than more recent ones) in order to balance the allocation.

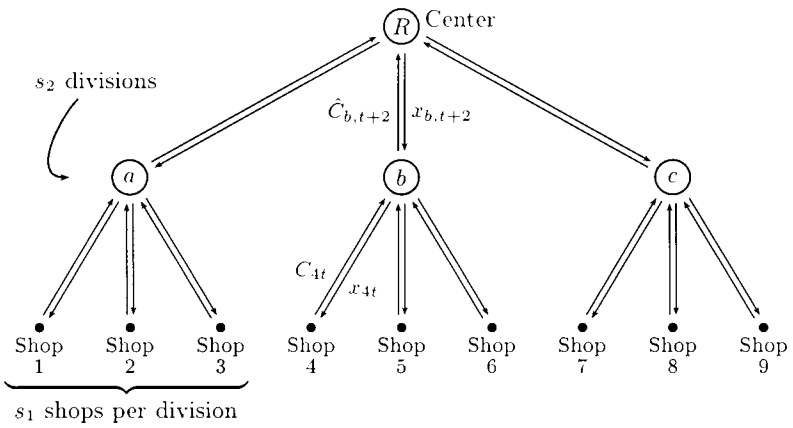

FIGURE 4. Information flow at the beginning of period t in a three-tier hierarchy.

$t+1, \ldots, t+d$. Although these computations overlap temporally, they are independent and do not use common data. As in Section 3.4, there is decentralized information processing within the central office because the procedure is computed in parallel, but decision making is not decentralized because, in each period, all the shops' allocations are computed from the same data. In particular, there is no hierarchical disaggregation of resource allocations.

4.2 A procedure with decentralized decision making

In this real-time model, we are not restricted to procedures such as those of Section 4.1, in which each period's decisions are calculated from data of the same lag. We now specify a procedure without this property.

We first present the hierarchical structure that we ascribe to the procedure. Assume there are integers $s_1, s_2 \geq 2$ such that $n = s_1 s_2$. Consider a balanced three-tier hierarchy in which the span of an office in tier $h \in \{1, 2\}$ is s_h. This means that the root (in tier 2) has s_2 immediate subordinates, which are offices in tier 1, and that each of these offices has s_1 immediate subordinates, which are shops in tier 0. There are thus $s_1 s_2 = n$ shops, as required. Such a hierarchy is shown for $n = 9$ and $s_1 = s_2 = 3$ in Figure 4.

In the decision procedure we describe, each office calculates resource allocations in much the same way as the center does in the procedure defined in Section 4.1. However, an office in tier 1 uses as its quantity of resources an amount that is sent by the center. The center uses as its cost information the aggregate cost functions calculated by the tier-1 offices. These aggregate cost functions are also used by the tier-1 offices to determine the suballocations of resources to their subordinate shops.

First, disregard the root of this hierarchy and imagine that we have simply divided the organization into s_2 independent units, which we call "divisions", even though they are independent. Each division contains s_1 shops. Because these divisions operate independently and resource allocations to the divisions are not coordinated, each division allocates a fixed fraction of the total resource. For example, if the allocations represent net trades, then the total amount of the resource available to the whole organization and to each division is 0. In any case, denote division j's available resource in period t by x_{jt}.

Each of these divisions allocates resources using a two-tier hierarchy as described in Section 4.1, but with s_1 shops rather than n shops. These calculations are shown on the left

	Division j	Center
Begin,...,end	$t-d_1,\ldots,t-1$	$t-2-d_2,\ldots,t-3$
Delay	$d_1 = 2+\lceil \log_2 s_1 \rceil$	$d_2 = 2+\lceil \log_2 s_2 \rceil$
Data	$\{C_{i,t-d_1}\}_{i\in\theta_j}$, x_{jt}	$\{\hat{C}_{j,t-d_2}\}_{j\in\Theta_R}$, x_{Rt}
Aggregation	$\hat{C}_{jt} := \bigoplus_{i\in\theta_j} C_{i,t-d_1}$	$\hat{C}_{Rt} := \bigoplus_{j\in\Theta_R} \hat{C}_{j,t-d_2}$
Shadow price	$p_{jt} := \hat{C}'_{jt}(x_{jt})$	$p_{Rt} := \hat{C}'_{Rt}(x_{Rt})$
Allocation	$x_{it} := C'^{-1}_{i,t-d_1}(p_{jt})$	$x_{jt} := \hat{C}'^{-1}_{j,t-d_2}(p_{Rt})$

TABLE 2. Calculations by the divisions and the center in the three-tier hierarchy.

side of Table 2. The delay is denoted by d_1 and is equal to $2+\lceil \log_2 s_1 \rceil$. The aggregate cost function and shadow price calculated by division j for the purpose of allocating resources in period t are denoted by $\hat{C}_{jt}$ and p_{jt} and are equal to $\bigoplus_{i\in\theta_j} C_{i,t-d_i}$ and $\hat{C}'_{jt}(x_{jt})$, respectively.

These divisions and their decision procedures correspond to the three subtrees under the center in Figure 4. The advantage of splitting the organization this way it that the delay $2+\lceil \log_2 s_1 \rceil$ of each division is smaller than the delay $2+\lceil \log_2 n \rceil$ of the unified two-tier organization in Section 4.1. However, gains from trade (coordination) between the divisions are lost.

To exploit some of these gains from trade, we add the central office that is shown as the root node in Figure 4. The center also uses a decision procedure like the two-tier procedure described in Section 4.2, but the center's immediate subordinates are the division offices instead of the shops. The data that the center uses are the aggregate cost functions $\{\hat{C}_{jt}\}_{j\in\Theta_R}$ of the divisions, which are partial results of the divisions' decision procedures. The center's delay is $d_2 \equiv 2+\lceil \log_2 s_2 \rceil$. Its calculations are shown on the right side of Table 2.

Let's consider more closely the data the center uses and the calculations it performs. They are more complicated to describe than for the offices in tier 1 because of the following complications, which can be verified in Table 1. On the one hand, each office j in tier 1 needs to know its period-t allocation x_{jt} two periods *before* period t, so that it has time to suballocate the resources before before t. On the other hand, each office in tier 1 finishes the calculation of $\hat{C}_{jt}$ two periods before period t. Hence, to calculate the allocation to the divisions for period t, the center begins in period $t-2-d_2$ and finishes at the end of period $t-3$. When the center begins calculating the period-t allocation in period $t-2-d_2$, the most recent aggregate cost functions calculated by the divisions are $\{\hat{C}_{j,t-d_2}\}_{j\in\Theta_R}$. These are the data the center uses to calculate the period-t allocation to the divisions. Hence, the aggregate cost function $\hat{C}_{Rt}$ that the center calculates in order to allocate the resource for period t is

$$\hat{C}_{Rt} = \bigoplus_{j\in\Theta_R} \hat{C}_{j,t-d_2} = \bigoplus_{j\in\Theta_R} \bigoplus_{i\in\theta_j} C_{i,t-d_1-t_2} = \bigoplus_{i=1}^{n} C_{i,t-d_1-d_2}\ .$$

The center then computes the shadow price $p_{Rt} := \hat{C}_{Rt}(x_{Rt})$ and sets x_{jt} so that $\hat{C}_{j,t-d_2}(x_{jt}) = p_{Rt}$.[7]

[7] The center must perform this last step using the same cost functions from which it computed $\hat{C}_{Rt}$.

This upward flow of information and downward flow of allocations is illustrated in Figure 4. At the beginning of period t, each shop i sends its current cost function C_{it} to its divisional office and receives its current allocation x_{it} from this office. At the same time, each division j sends $\hat{C}_{j,t+2} = \bigoplus_{i \in \theta_j} C_{i,t+2-d_1}$ (a partial result from its computation of the period-$(t+2)$ allocation) to the center, and the center sends $x_{j,t+2}$ (the amount of the resource division j will allocate in period $t+2$) to each division j.

For example, toward the end of August a division office finishes aggregating information about its immediate subordinates' resource needs and sends this information to its immediate superior. At the same time, it receives a budget for September, which in the next few days it disaggregates in order to assign a September budget to each of its subordinates. The center calculated the division's September budget using information sent by that division and the other divisions at the end of July, but the division office disaggregates the budget using more recent information, which it aggregated during August.

4.3 The benefits and costs of decentralized decision making

In Section 4.2, we presented the three-tier hierarchy as a comparison of (a) independent two-tier hierarchies between which there is no coordination and (b) a three-tier hierarchy within which these two-tier hierarchies are coordinated by a central office. Now we compare (c) a two-tier hierarchy that has no hierarchical disaggregation and (d) a three-tier hierarchy with the same number of shops. That is, we compare Figures 3 and 4. In order to simplify this comparison, assume that s_1, s_2, and n are powers of two, so that the round-up operations in the formulas or the delay in the previous sections can be ignored. (Otherwise, some of the quantities in what follows could differ by 1.)

It may appear that one difference between the two-tier and the three-tier hierarchies is that, in the latter, the aggregation of cost functions is hierarchically decomposed and hence more decentralized. However, remember that each node in these hierarchies is an office containing multiple information processing agents, and that the aggregation of cost functions is always maximally decentralized within each node. This means that the hierarchical decomposition of the aggregation of cost functions, which is explicit in Figure 4, exists also within the center in Figure 3. Observe in particular that, in a three-tier hierarchy, the center finishes calculating in period $t-4$ the aggregate cost function $\hat{C}_{Rt}$ that it uses to allocate resources for period t. Furthermore, $\hat{C}_{Rt} = \bigoplus_{i=1}^{n} C_{i,t-d_1-d_2}$. Hence, the aggregate cost function is calculated in

$$d_1 + d_2 - 4 \;=\; (2 + \log_2 s_1) + (2 + \log_2 s_2) - 4 \;=\; \log_2 n$$

periods in the three-tier hierarchy. This is exactly the same delay as in the two-tier hierarchy.

The actual difference between the two-tier and the three-tier hierarchies is the disaggregation of resource allocations. The center in a three-tier hierarchy, after calculating its aggregate cost function $\hat{C}_{Rt}$ and then its shadow price p_{Rt}, does not allocate resources directly to the shops in one step. Instead, it calculates allocations for the divisions, whose offices then calculate suballocations for the shops. We interpret this as decentralized decision making, as explained in Section 5. The advantage of this may be summarized as follows.

That is, it must set x_{jt} so that $\hat{C}'_{j,t-d_2}(x_{jt}) = p_{Rt}$, even though the divisions have calculated more recent aggregate cost functions (e.g., $\hat{C}_{j,t-3}$); otherwise the allocation will not be balanced.

Benefit	2-tier: Delay	$2 + \log_2 n$
	3-tier: Delay of each division	$2 + \log_2 s_1$
	Diff: **Decrease** in division's delay ..	$\log_2 s_2$
Cost	2-tier: Delay	$2 + \log_2 n$
	3-tier: Center's cumulative delay	$(2+\log_2 s_1) + (2+\log_2 s_2)$
	Diff: **Increase** in center's delay	2
Cost	2-tier: Operations of center	$2n$
	3-tier: Operations center & divisions ..	$2s_2 + s_2(2s_1)$
	Diff: **Increase** in operations	$2s_2$

TABLE 3. The benefits and costs of decentralized decision making (three-tier versus two-tier hierarchies).

When a division receives x_{jt} from the center in period $t-2$, it does not have to allocate x_{jt} using the information $\hat{C}_{j,t-d_2}$ that the center used to compute x_{jt}. Instead, it uses its most recently calculated aggregate cost function $\hat{C}_{jt}$. That is, *the data used to allocate resources within each division in a three-tier hierarchy are* $2 + \log_2 s_1$ *periods old, and hence* $\log_2 n - \log_2 s_1 = \log_2 s_2$ *periods more recent than the data used to compute allocations in the two-tier hierarchy.*

The intermediate disaggregation of resource allocations adds two extra steps: the calculation of the divisions' allocations and the calculation of the divisions' shadow prices. This leads to two disadvantages of the three-tier hierarchy compared to the two-tier hierarchy.

1. In the three-tier hierarchy, gains from trade between shops in different divisions are exploited by the center. The cumulative lag of the center's data that it uses to allocate resources is

$$d_1 + d_2 = (2 + \log_2 s_1) + (2 + \log_2 s_2) = 4 + \log_2 n .$$

This is two periods greater than the center's lag in the two-tier hierarchy. This extra lag is a *decision-theoretic cost* of decentralized decision making.

2. The three-tier hierarchy also has higher managerial costs. The center's calculations in the three-tier hierarchy involve $2s_2$ operations per period. Each division's calculations involve $2s_1$ operations per period. Hence, the total number of operations is $2s_2 + s_2(2s_1) = 2s_2 + 2n$. In contrast, the number of operations in the two-tier hierarchy is only $2n$. The wages paid for the $2s_2$ additional operations in the three-tier hierarchy are a *managerial cost* of decentralized decision making.

The benefits and costs are summarized in Table 3.

We have now described real-time decision procedures that correspond to two-tier hierarchies and to three-tier hierarchies that are balanced (each node in the same tier has the same number of subordinates). In the Appendix, we generalize the real-time procedures to arbitrary hierarchies. Increased decentralization of decision making corresponds to additional intermediate tiers. The benefits and costs of three-tier versus two-tier hierarchies arise again as additional tiers are added.

5 Interpretation

Recall that the nodes in the three-tier hierarchy are offices rather than individual managers. Because our model has no communication costs, we cannot even state that the agents who perform operations within one node at one point in time must be different from the agents who perform operations within a different node at another point in time.[8] Hence, the hierarchical structure we see in Figure 4 arises from the structure of the decision procedure, rather than from communication between individual administrators.

This a realistic view of organizations. An organizational chart does not show the links between every manager, professional, secretary, clerk, and computer in an administrative apparatus. Instead, it shows offices within which many people and machines may work. Furthermore, the chart depicts the structure of decision-making procedures that persist over time, even when there are changes in personnel. Such changes occur not only when someone retires or finds a new job, but also when an employee is temporarily absent and either a new employee is hired as a substitute or an existing employee in another office fills in. Some employees even spread their time on a regular basis between two positions in different offices. Hence, whereas the literature on organizations that process information with an endogenous number of agents has focused on the micro structure of communication between individual agents,[9] we should be at least as interested in the macro structure of communication between offices and division nodes.

We claim that the intermediate disaggregation of allocations in the three-tier hierarchy (and in the general hierarchies defined in the Appendix) is related to decentralized decision making. This claim is considered more formally in Van Zandt (1998c); here we limit ourselves to an informal interpretation.

Decentralized decision making does not have a universal formal definition. The reader may associate decentralized decision making with the delegation of certain decisions to specific individuals who are autonomous in some sense and who may even have conflicting interests. However, there are no conflicts of interest in this model, and the model could not formalize the notion of autonomy. Furthermore, we have just explained why calculations or decisions in this model should not be identified with individuals.

We instead look to the definition of decentralized decision making that is implicit in team theory (Marschak and Radner (1972)). Accordingly, there is decentralization of decision making when individuals or offices (i) control different action variables and (ii) base their decisions on different information (Radner (1972b, p. 189)).

The first criterion is not entirely precise. In our resource allocation problem, are the only action variables the allocations of the individual shops? Can we consider the aggregate allocations of a division, which are not intrinsic control variables in the decision problem, to be action variables? Should every partial result of the calculations be considered an action variable, given that its value ultimately influences the final allocations? Our answers to these questions are, respectively, "no", "yes", and "no", but our justification is not formal. Rather, the decision procedures in our model do resemble actual hierarchical budgeting or other resource allocation procedures, and most observers would classify the resource

[8] That same indeterminateness, on the other hand, means that it is *possible* to assign each operation within each node (which is repeated every period) to the same agent. One could imagine advantages to doing so given positive communication costs or the kinds of returns to specialization studied in Bolton and Dewatripont (1994), but this would take us outside our model.

[9] For example, Geanakoplos and Milgrom (1991), Radner (1993), and Bolton and Dewatripont (1994); an exception is Reiter (1996)

allocations that come out of offices as decisions but would not classify every partial result of the calculations as a decision. Perhaps this is because the aggregate allocation that an office transmits to a subordinate division constrains the possible resource allocations for that division. Therefore, we view the joint information processing within each office as decentralized information processing but not as decentralized decision making.

The second criterion, which has been called *informational decentralization* (e.g., Radner (1972a, p. 188)), is critical for a model in which there are no conflicts of interest, because otherwise it makes no difference who controls which actions. However, the meaning of criterion (ii) and its relationship to decentralized decision making is also imprecise. Is there informational decentralization if the decisions made are the same as those that would have been made had all information been shared? Take, for example, iterative planning procedures and static communication mechanisms that include exchanges of demand or price information in a space of lower dimension than the initial private information; if the allocations or outcomes thereby calculated are the same as would have been realized by a full exchange of information, is this then an instance of decentralized decision making? Without taking a stand on this semantic question, we at least can say that the sense of decentralization is stronger when the decisions that are taken are not the same as would be taken with a full exchange of information.

In our model, this informational decentralization is present and exactly matches the hierarchical structure. In the two-tier hierarchy, all allocations at time t are a function of $\{C_{i,t-d}\}_{i=1}^{n}$. In the three-tier hierarchy, the allocations at time t to the shops in division j are a function of the data $\{C_{i,t-d_1-d_2}\}_{i=1}^{n}$ that the center implicitly uses to compute x_{jt} and of the data $\{C_{i,t-d_1}\}_{i\in\theta_j}$ that division j uses to suballocate x_{jt}. This information is different for shops in different divisions. Furthermore, the allocations to the subordinates of any one node of the hierarchy are functions of the same data, whereas the allocations to the subordinates of different nodes are functions of different data.[10]

This is why we consider the three-tier hierarchy to have more decentralized decision making than the two-tier hierarchy, even though both hierarchies have maximally decentralized information processing. Note that we are simply comparing two- and three-tier hierarchies and are not claiming that the three-tier hierarchy has the greatest possible decentralization of decision making. For example, hierarchies with more tiers are defined in the Appendix. There also exist nonhierarchical procedures that are less bureaucratic and that more closely resemble market mechanisms for allocating resources. All of these may be more decentralized than three-tier hierarchies.

6 Related literature on decentralization

Recall that our model simultaneously explains (a) why agents with no *prior* private information are hired to process information and (b) why decision making is then decentralized among them. Both effects are due solely to information processing delay, as is highlighted by our model's total lack of information transmission costs and incentive problems. Here we consider other explanations for either (a) or (b).

[10]Note that the procedures involve the calculation of shadow prices, but different offices in the three-tier hierarchy calculate distinct shadow prices at any point in time. Hence, the decentralization that arises because of information processing constraints causes a failure of the law of one price.

6.1 Information transmission costs

In most of the preceding economics literature on decentralized decision making—such as the iterative planning, message space, and team theory literatures—decentralization has been due to information transmission costs. (See Van Zandt (1998a) for a brief survey.) The intuition is straightforward. If agents (such as the shops in our model) have heterogeneous information and it is costly for them to share this information, the decisions should be delegated to the agents with the best information for that decision.

The delegation of decisions to intermediaries (such as to the administrators in our model) who are not exogenously endowed with private information would only increase transmission costs. However, in a model with both computation constraints and transmission costs, the former can motivate the hiring of agents to process information, and then—since these agents thereby acquire private information—the latter may provide additional motivation for decentralizing decision making to those agents. For example, as described in Bernussou and Titli (1982, p. 25), if the shops are distributed spatially and the transmission of information over long distances is costly, then it may reduce communication costs to have local management units that coordinate small groups of neighboring shops along with a central management unit that coordinates the local units, as opposed to having all management activities in a single location. Mount and Reiter (1990) develop a general computation model that incorporates communication constraints, which, as applied in the batch processing model of Reiter (1996), yields examples of decentralized decision making.

6.2 Incentives

Incentive problems tend to work against decentralization. Like communication costs, they are aggravated by the hiring of information processing intermediaries (because these intermediaries may shirk and obtain rents from private information). Furthermore, according to the revelation principle, pure incentive problems are not lessened and may be aggravated by decentralizing decision making to agents whose private information is exogenously given. Nevertheless, there have been a few incentives-based justifications for decentralization, which we now describe. (Section 7.3 mentions other papers that take decentralization as given and consider its consequences for incentives.)

There are models in which outside agents are hired to monitor effort or audit the output of workers and perhaps each other (e.g., Baiman et al. (1987), Baron and Besanko (1984), Calvo and Wellisz (1980), Demski and Sappington (1987), and Qian (1994)), but these agents are purely sources of information and there is no delegation of decision-making tasks to them. In models with hidden information, the revelation principle continues to apply with respect to the entire set of agents and auditors, so that the auditors also communicate directly with the principal and not with the other agents.

Several papers have shown—in models with two or three parties who are endowed with private information—that decentralizing decision making can strictly dominate centralization when complete, enforceable contractual mechanisms are not possible because of post-contracting collusion, renegotiation, or lack of commitment. (a) Collusion. Laffont and Martimort (1997) find, in an adverse selection model, that collusion does not by itself favor decentralization (there are optimal centralized collusion-proof mechanisms). Similarly, Baliga and Sjöström (1998) is a moral hazard model in which agents can collude and delegation changes the division of surplus by changing the agents' outside options, yet decentralization only weakly dominates centralization. However, Laffont and Martimort

(1998) find that delegation can be strictly optimal under the assumptions that agents may collude, that communication is constrained, and that delegation changes (i) the division of surplus between the agents when they bargain and (ii) the timing of individual rationality constraints. (b) Renegotiation. In Poitevin (1995), delegation of decision making may be strictly optimal because it is assumed to limit renegotiation (by reducing the level of contractually stipulated communication). (c) Lack of commitment. When there is a lack of commitment over observable variables, decision-making authority should be assigned taking into account the parties' ex-post incentives. For example, the assignment of property rights in the incomplete contracts literature can be interpreted as delegation of decision-making authority. Klibanoff and Poitevin (1995) find that it may be strictly optimal for a principal to delegate decision making to agents in a model in which the principal lacks commitment and there are externalities between the agents.

6.3 Information processing

The batch processing literature has modeled the endogenous hiring of information processing agents in large administrative apparatus in order to reduce delay or increase throughput. However, in Section 3, we showed that a benchmark batch processing model did not demonstrate any advantage to decentralized decision making. This exercise illustrated that the decomposition of decision problems into steps that are performed in parallel by multiple agents is not the same as decentralized decision making, and hence it is not trivial that constraints on individual information processing capabilities lead to the latter. It was not a claim that only in a real-time processing model (or only because of delay) could bounded rationality lead to decentralized decision making. For example, the literature on multi-level systems in operations research and management science (e.g., Bernussou and Titli (1982), Dirickx and Jennergren (1979), and Dudkin et al. (1987)) mentions a variety of unquantified but intuitive advantages to decentralized decision making, distinct from the one presented in this paper.

Another example is Geanakoplos and Milgrom (1991), who present a model in which bounded rationality also leads to hierarchically decentralized decision making in a resource allocation problem. Although it is a static team theory model in which delay plays no specific role, it is an important tool in Van Zandt (1998c) for defining organizational structures and deriving real-time processing procedures that take into account the team statistical inference problem. In their model, the aggregation of information is not modeled. Instead, manager acquire information directly from the environment and bounded rationality takes the form of constraints on the amount of information each manager can acquire. One consequence is that it does not have an endogenous formulation of the differences between information at different nodes of the hierarchy. At an informal level, our model can motivate their assumption that aggregate information is less accurate than disaggregate information, but their model is not a reduced form of Van Zandt (1998c), as explained there and in Van Zandt (1998e).

Radner and Van Zandt (1992) and Van Zandt and Radner (1998), which study real-time processing and returns to scale of firms, are also implicitly about decentralized decision making. However, their decentralization takes a stark form—it occurs when decision problems are divided up into completely separate units between which there is no coordination. In contrast, this paper models decentralized decision making within unified non-market organizations.

7 Further comments and extensions

7.1 Optimality

Q1. Should we care about optimality?

Introducing information processing constraints into a decision problem such as the one studied in this paper restricts the set of feasible decision rules and also adds information processing costs that are part of the performance criteria. *Constrained optimality* can be defined to be "optimality given the information processing constraints". As is obvious, we do not characterize constrained-optimal decision procedures in this paper. However, we do compare the performance of different classes of decision procedures, and the motivation for doing so is no different than the motivation for characterizing constrained-optimal procedures. This motivation is discussed in Van Zandt (1998b, Section 2.3), where it is explained why—even as a descriptive criteria—constrained optimality is consistent with the bounded rationality of the agents in this model and does not presume that these agents or any others can effortlessly and instantly design constrained-optimal organizations.

Q2. Are the procedures described in Section 4 constrained-optimal?

There is not enough structure in this paper to measure the expected value of shop costs for different procedures, so we cannot even rank any two procedures in terms of constrained optimality. All we have shown is that there is a potential advantage of decentralized hierarchical procedures over centralized hierarchical procedures. The question of whether the procedures are constrained-optimal is addressed more meaningfully in Van Zandt (1998c, 1998e), where it is possible to calculate the sum of the shop and administrative costs for given procedures.

However, note that we neither claim nor expect that, under most statistical assumptions, the procedures in Section 4 and their generalization to arbitrary hierarchies are better than others not considered here. Even in this very abstract version of the model, which has a coarse set of elementary functions, the set of possible procedures for calculating resource allocations is huge. Minor variations that could be better or worse than the procedures we study include (a) not updating the resource allocations each period or never processing cost information about some shops or divisions to whom resources are allocated (in order to reduce information processing costs), and (b) taking into account the stochastic processes governing the evolution of the cost functions, as in Van Zandt (1998c)) (rather than calculating resource allocations that are optimal given old cost functions). We describe more significant variations in Section 7.2.

7.2 Alternative procedures

Q3. How could we model a market mechanism?

It is possible to model market mechanisms as alternate decision procedures within our model, thereby comparing the computational efficiency of markets with the computational efficiency of hierarchies. This would provide a formal analysis of why certain transactions take place in markets and others take place in hierarchies. This would also be an extensive project, but we can give a simple example here. It illustrates the versatility of the general methodology of modeling information processing in organizations as real-time control.

As a first step, we can assume that, even when the agents interact in markets, they do so with the objective of minimizing collective costs; the incentive structure is thereby the same in the two models. That is, we can collectively design the market interaction and the decision procedures that each agent uses in the markets. Although this is a rather artificial assumption, it does allow us to focus on computational differences between markets and bureaucracies, which should be done before we study these differences in combination with differences in incentives.

There are many market mechanisms and structures, such as auctions, wholesale/retail networks, and financial specialists. Some of these even have semihierarchical structures. Perhaps the most decentralized market interaction is bilateral trade, and so we use this for our example.

We now think of each shop as the decision-making unit. The shop may still be an "office"; for comparison with the hierarchical procedures, we should allow for maximal decentralization of the operations that the shop must perform. We assume that, in d_m cycles, all shops can be (randomly or deterministically) matched pairwise. The shops keep track of their current allocation, which they do not change until they calculate a new one through a pairwise exchange. Once matched, each pair of shops calculates an exchange (so that their total resources do not change) that minimizes the sum of their costs. This can be done with the elementary operations we have already defined. We do not bother to specify the details of this calculation here; let's simply say that it takes d_e cycles. Then, every $d_m + d_e$ cycles, shops are matched and within each pair the allocations are updated based on information that is d_e periods old.

Here is one comparison between this bilateral exchange and the hierarchical procedures. First, observe that we could generalize this procedure so that, instead of pairwise matches, trade takes place within groups of size s_1; thus, when $s_1 = 2$ we have the bilateral model. (We are deliberately using the same symbol s_1 that we used for the size of each division in the three-tier hierarchies.) Suppose that, as an alternative to the two-tier hierarchy in Section 4.1, we simply split the organization up into units of size s_1, so that resource allocations within the units are based on more recent information but gains from trade between units are not exploited. In Section 4.2, to take advantage of these gains from trade we added a center that coordinated trades between the units. Now suppose that instead we simply mix up the units over time as in the pairwise matching. Then we no longer have fixed groups that never trade even when the marginal costs for the groups become very different. Hence, the trading within small groups combined with mixing of the groups allows resource allocations to be computed from recent information while at the same time taking advantage of gains from trade across the population.

Q4. Can iterative procedures be modeled?

Friedman and Oren (1995) study a batch processing model for the resource allocation problem without externalities, a model in which the algorithm is an iterative procedure similar to a Walrasian tatônnement or iterative price–quantity planning process. Thus, its set of elementary operations does not include the very unelementary operation $\div$ that appears in the current paper. The purpose of their paper is to measure the parallel complexity of the resource allocation problem.

We can also define hierarchically decentralized iterative procedures that are direct analogs to the centralized and decentralized hierarchical procedures in Section 4. This indicates that the main message of this paper does not depend critically on our particular decomposition of the decision problem into elementary operations.

In our model, global cost information flows up the hierarchy and resource allocations flow down. The analog in an iterative procedure is that local cost information, in the form of marginal costs (shadow prices), flows up and is aggregated by the hierarchy, and resource allocations flow down. This corresponds to a real-time version of a gradient ascent algorithm that is the basis of the quantity–price planning procedure in Heal (1969). In the iterative procedures, each office allocates resources based on aggregated marginal costs (average shadow prices) of the shops inferior to the offices, and this information is different for different nodes of the hierarchy. An example of such procedures is defined for general hierarchies at the end of the Appendix.

Defining multitier real-time versions of the more classical price–quantity Walrasian procedure in Friedman and Oren (1995) poses several difficulties, which we have not yet thought through. In such a model, demands at each point in time would be aggregated through the hierarchy and shadow prices would be transmitted and updated down the hierarchy. However, it is impossible for the organization to satisfy a binding budget constraint with this type of model. Furthermore, there is no analog to the *disaggregation* of shadow prices, although it is possible for each office to update the shadow price it receives from its immediate superior before sending the shadow price on to the subordinates.

7.3 Complications

Q5. What if there are also incentive problems?

The bounded rationality approach and the incentive approach to the economics of organizations have so far developed independently, but it has long been recognized that incentive problems and bounded rationality interact strongly in organizations.

There are two classes of such interaction. First, incentive problems require contracts or mechanisms whose clauses must be stated ex ante and calculated ex post. This introduces costs and constraints on the set of feasible contracts and mechanisms, and it creates information processing tasks that cannot be delegated directly to interested parties without creating further incentive problems. The contracting contraints in the incomplete contracts literature are often informally motivated by complexity and bounded rationality. Williams (1986), Reichelstein and Reiter (1988), and Hong and Page (1994) have studied mechanism design explicitly taking into account communication costs (which may be motivated by bounded rationality).

Second, the delegation of any information processing tasks creates problems of private information and, if the effort exerted by the information processing agents exert is not observable, of moral hazard. This interaction has been studied implicitly in the and the hierarchical contracting literature, such as McAfee and McMillan (1995), Melumad et al. (1992, 1995, 1997), and Mookherjee and Reichelstein (1995, 1997)). These papers start with standard adverse selection models in which the revelation principle would apply, but then impose hierarchical decentralization of contracting, motivated informally by bounded rationality or explicitly by communication costs. They then characterize the optimal ("third-best") contracts, and determine when there is a strict efficiency loss due to decentralization (compared to the direct mechanism benchmark).

Models of information processing such as the one in this paper can be used to study these issues formally. For example, it might be possible to integrate our model, framed as a profit maximization problem with managers deriving positive benefit from being allocated resources, with the hierarchical contracting models mentioned above, several of which are also based on resource allocation problems. Conceivably, incentives could provide

Nodes:

J	=	set of offices (internal nodes).
$\{1,\ldots,n\}$	=	set of shops (terminal nodes).
R	=	root ($R \in J$).
H	=	height

Subordinates: Define for each office $j \in J$.

Θ_j	=	set of direct subordinates of office j:
s_j	=	span of office j ($s_j = \#\Theta_j$).
θ_j	=	set of shops inferior to j (division j). (For each shop i, let $\theta_i = \{i\}$.)
n_j	=	size of division j ($n_j = \#\theta_j$). (For each shop i, let $n_i = 1$.)

TABLE 4. Notation for general hierarchies.

an additional reason for decentralizing decision making. Specifically, it may be that the computation constraints motivate the hiring of agents to process information, whereupon it becomes easier to measure the performance of agents and to control against improper use of resources by hierarchically decomposing the disaggregation of allocations.

Q6. What if there are externalities?

If the environment exhibits externalities—meaning that each cost function depends (potentially) on the entire vector of allocations—then the decomposition in Section 2 is not possible. Even the centralized procedure, which relied on the operation $\oplus$, would have to change in the presence of externalities. In these brief comments, we can at best speculate on the properties of real-time resource allocation with externalities.

It would still be possible to define hierarchically decomposed real-time procedures, but each office would not be able to fully take into account externalities between shops in its own division and other shops in the organization. This would be an additional disadvantage of decentralization. On the other hand, the advantage of decentralization described in Section 4.2 would still be present. It should be possible to construct a model of this trade-off such that, for some parameter values (e.g., when externalities are not too significant), it is better to decentralize the decision making in order to take advantage of the reduced delay of lower levels of the hierarchy, whereas for others (e.g., when externalities are important and the environment does not change too quickly), centralized procedures perform better.

Appendix: General hierarchies

In this appendix, we define a real-time procedure for an arbitrary hierarchy such that the procedure is analogous to the ones in Section 4 and the decomposition of the resource allocation problem for the hierarchy is as defined in Section 2.

Recall the notation from Section 2 that we used to describe a hierarchy. It is listed in Table 4, along with the following. The *span* s_j of division j is the number of direct subordinates in Θ_j, and the *size* n_j of division j is the number of shops in θ_j. As a

Periods $(t-\tau_j-\cdots)$	Calculation	Operations Type	#	Delay
$d_j,\ldots,3$	$\hat{C}_{jt} := \bigoplus_{k\in\Theta_j} \tilde{C}_{kt}$	f_1	$s_j - 1$	$\lceil \log_2 s_j \rceil$
2	$p_{jt} := \hat{C}'_{jt}(x_{jt})$	f_2	1	1
1	$x_{kt} := \tilde{C}'^{-1}_{kt}(p_{jt})$	f_3	s_j	1
		Total:	$2s_j$	$\lceil \log_2 s_j \rceil + 2$

$\tau_j = 2(h_j - 1)$
$d_j = \lceil \log_2 s_j \rceil + 2$
$\tilde{C}_{kt} = \hat{C}_{k,t-d_j-2(h_j-h_k-1)}$

TABLE 5. The calculations performed by office $j \in J$ for the allocation of resources in period t.

convention, we define $\theta_i = \{i\}$ and $n_i = 1$ for each shop i. The *tier* h_k of node k is the maximum of the lengths of the paths from the node to shops in θ_k. Hence, all the shops are in tier 0 and the offices are in higher tiers. The tier $H \equiv h_R$ of the root is called the *height* of the hierarchy and is the number of managerial tiers. For $k_1, k_2 \in I \cup J$, we write $k_1 \gtrsim k_2$ if k_2 is weakly inferior to k_1 in the hierarchy.

Note that $h_j = 1 + \max\{h_k \mid k \in \Theta_j\}$ for $j \in J$. If $k \in \Theta_j$ and $h_k + 1 < h_j$, then k is said to skip $h_j - h_k - 1$ levels when reporting to j, or simply to "skip-level report". For example, in Figure 1, manager d skips one level when reporting to R and all other managers do not skip-level report. The balanced three-tier hierarchy in Section 4.2 has no skip-level reporting.

The calculations performed by office $j \in J$ for the period-t allocation are shown in Table 5; they are a generalization of the calculations in a three-tier hierarchy. Office $j \in J$ begins calculating the period-t allocation of its subordinates in some period before t (yet to be determined) by collecting the latest aggregate cost function calculated by each of its subordinates (or just the subordinate's latest cost function, if the subordinate is a shop). Let $\tilde{C}_{kt}$ be this cost function for subordinate $k \in \Theta_j$, where the t denotes the period for which j uses the information to allocate resources (rather than the period in which $\tilde{C}_{kt}$ is collected). Office j then calculates its own aggregate cost $\hat{C}_{jt} = \bigoplus_{k\in\Theta_j} \tilde{C}_{jt}$ in $\lceil \log_2 s_j \rceil$ periods from this data; the index t in $\hat{C}_{jt}$ again denotes the period of the allocation for which the information is used. In the next two periods, j calculates its period-t shadow price $p_{jt} = \hat{C}'_{jt}(x_{jt})$ and then the allocations $\left\{x_{kt} = \tilde{C}'^{-1}_{kt}(p_{jt})\right\}_{k\in\Theta_j}$ of its subordinates. Office j's delay is thus $d_j \equiv \lceil \log_2 s_j \rceil + 2$, and it performs $2s_j$ calculations per period.

A subtlety that complicates description of the timing and the use of data is that each office must finish calculating the period-t allocation of its immediate subordinates two periods *before* any such subordinate $k \in \Theta_j$ (if it is an office) finishes calculating the period-t allocation of k's own immediate subordinates (because this is when k uses x_{kt} as an input, as seen in Table 5). It follows by induction that the latest time each office j can finish calculating the period-t allocation is by the end of period $t - \tau_j - 1$, where $\tau_j = 2(h_j - 1)$. (If subordinate $k \in \Theta_j$ does not skip-level report to j, then k learns x_{kt} exactly when it needs this information; otherwise, k learns it $2(h_j - h_k - 1)$ periods earlier than necessary.) So that this is the period in which j finishes, we let j begin calculating the period-t allocation in period $t - \tau_j - d_j$.

Next we determine what the latest available cost function $\tilde{C}_{kt}$ is for $k \in \Theta_j$ when office j collects $\left\{\tilde{C}_{kt}\right\}_{k\in\Theta_j}$ at the beginning of period $t - \tau_j - d_j$. If k is an office then, for each $t' \in \mathbb{Z}$, k finishes calculating the period-t' allocation of its own immediate subordinates at the end of period $t' - \tau_k - 1$. An intermediate result of this calculation is $\hat{C}_{kt'} = \bigoplus_{l\in\Theta_k} \tilde{C}_{lt'}$, which (from Table 5) is available two periods earlier, or at the beginning of period $t' - \tau_k - 2$. If k is an shop then we can also define $\tau_k = 2(h_k - 1) = -2$ (since $h_k = 0$) and we can define $\hat{C}_{kt'} = C_{kt'}$; hence it is also the case that $\hat{C}_{kt'}$ is available at the beginning of period $t' - \tau_k - 2 = t'$. Therefore, whether k is an office or a shop, $\tilde{C}_{kt} = \hat{C}_{kt'}$ for t' such that

$$\begin{aligned} t' - \tau_k - 2 &= t - \tau_j - d_j \ , \\ t' - 2(h_k - 1) - 2 &= t - 2(h_j - 1) - d_j \ , \\ t' &= t - d_j - 2(h_j - h_k - 1) \ . \end{aligned}$$

Let

$$L_{jk} = d_j + 2(h_j - h_k - 1) \ ,$$

which is the lag of the data about k used by j (i.e., $\tilde{C}_{kt} = \hat{C}_{k,t-L_{jk}}$). Observe that if k does not skip-level report then $h_j - h_k - 1 = 0$ and $L_{jk} = d_j$. That is, these various timing complications cancel and the lag of the data used by j is equal to j's delay.

As we move up a hierarchy, each office adds an extra lag to the data due to computational delay. For $k_1, k_2 \in I \cup J$ such that $k_1 \succsim k_2$, we define the *cumulative lag* $L_{k_1k_2}$ of k_1's information about k_2 to be the sum of $L_{l_1l_2}$ for $l_1 \in J$ and $l_2 \in \Theta_j$ such that l_1 and l_2 are on the path from k_1 to k_2 in the hierarchy:

$$L_{k_1k_2} \equiv \sum_{\substack{l_1\in J,\ l_2\in\Theta_{l_1}: \\ k_1\succsim l_1\succ l_2\succsim k_2}} L_{l_1l_2} \ . \tag{1}$$

Note that we have not redefined $L_{k_1k_2}$ if $k_1 \in J$ and $k_2 \in \Theta_j$, and that if $k_1 = k_2$ then $L_{k_1k_2} = 0$. Furthermore, if $k_1, k_2, k_3 \in I \cup J$ and $k_1 \succsim k_2 \succsim k_3$, then $L_{k_1k_3} = L_{k_1k_2} + L_{k_2k_3}$.

We now obtain the following characterization of the information on which each office's allocation decisions are based.

Proposition 1 *For $j \in J$, $\hat{C}_{jt} = \bigoplus_{i\in\theta_j} C_{i,t-L_{ji}}$.*

PROOF: The proof is by induction on the tier of j.

If $h_j = 1$, then all of j's immediate subordinates are shops ($\Theta_j = \theta_j$); hence $\hat{C}_{jt} = \bigoplus_{k\in\Theta_j} \hat{C}_{k,t-L_{jk}} = \bigoplus_{i\in\theta_j} C_{i,t-L_{jk}}$.

Let $h \in \{1, \ldots, H-1\}$ and assume that $\hat{C}_{jt} = \bigoplus_{i\in\theta_j} C_{i,t-L_{ji}}$ for $j \in J$ such that $h_j \leq h$. Let $j \in J$ be such that $h_j = h + 1$. Then

$$\hat{C}_{jt} = \bigoplus_{k\in\Theta_k} \hat{C}_{k,t-L_{jk}} = \bigoplus_{k\in\Theta_j} \left(\bigoplus_{i\in\theta_k} C_{i,t-L_{ki}-L_{jk}}\right) \ .$$

Since $L_{ki} + L_{jk} = L_{ji}$ and since $\{\theta_k\}_{k\in\Theta_j}$ is a partition of θ_j, it follows that $\hat{C}_{jt} = \bigoplus_{i\in\theta_j} C_{i,t-L_{ji}}$. □

We can also decompose the cumulative lag as follows.

Proposition 2 *For $j \in J$ and $k \in I \cup J$ such that $j \succ k$,*

$$L_{jk} = 2(h_j - h_k) + \sum_{\substack{l\in J \\ j\succsim l\succ k}} \lceil \log_2 s_l \rceil \ . \tag{2}$$

The summation is the cumulative aggregation delay and $2(h_{k_1} - h_{k_2})$ is the cumulative disaggregation delay.

PROOF: Since $L_{jk} = d_j + 2(h_j - h_k - 1)$ and $d_j = \lceil \log_2 s_j \rceil + 2$ for $j \in J$ and $k \in \Theta_j$, we can rewrite (1) as

$$L_{k_1 k_2} = \sum_{\substack{l_1 \in J,\ l_2 \in \Theta_{l_1}: \\ k_1 \succsim l_1 \succ l_2 \succsim k_2}} \lceil \log_2 s_{l_1} \rceil + 2(h_{l_1} - h_{l_2}) \ .$$

In the summation of the terms $h_{l_1} - h_{l_2}$, intermediate terms cancel, leaving $2(h_{k_1} - h_{k_2})$. Hence,

$$L_{k_1 k_2} = 2(h_{k_1} - h_{k_2}) + \sum_{\substack{l_1 \in J: \\ k_1 \succsim l_1 \succ k_2}} \lceil \log_2 s_{l_1} \rceil \ ,$$

which is the same as equation (2) for $j = k_1$ and $k = k_2$. □

Proposition 3 *The total number of operations per period is* $2(n + \#J - 1)$.

PROOF: From Table 6, office $j \in J$ has $2s_j$ operations per period. Hence, the total number of operations is $\sum_{j \in J} 2s_j$. Each shop and each office (other than the root) is the subordinate of one and only one office, and the root is not a subordinate of any office. Hence, $\sum_{j \in J} s_j = n + \#J - 1$. □

The costs of decentralization are (a) 2 operations per period per additional office and (b) 2 periods of cumulative delay per tier. The benefit of decentralization—that is, the benefit of having an office $k \in J \setminus \{R\}$ compared to eliminating that office and having its immediate superior j allocate resources directly to k's immediate subordinates—is that k suballocates the resources using information about the shops inferior to k that is more recent than the information used by j if k is eliminated.

We conclude by defining an iterative analog of these general hierarchical procedures, as described in Section 7.2. The elementary operations are arithmetic and differentiation (f_2). We assume that each arithmetic operation takes one period and that differentiation takes d_0 periods. We treat the shops as offices that process information because the calculation of marginal costs (shadow prices) from cost functions is performed only at the level of the shops in any hierarchy. By the beginning of each period t, each shop i finishes calculating $\hat{p}_{it} \equiv p_{i,t-d_0} \equiv C'_{i,t-d_0}(x_{i,t-d_0})$. The operations performed by each managerial office $j \in J$ in the hierarchy are shown in Table 6. The hierarchy recursively aggregates the shadow prices and disaggregates the resource allocations.

Compare a two-tier and three-tier hierarchy. In a two-tier hierarchy, allocations are updated using cost information (marginal cost) that is $d_0 + \lceil \log_2 n \rceil + 5$ periods old. In a balanced hierarchy, allocations within each division are updated using cost information that is $d_0 + \lceil \log_2 s_1 \rceil + 6$ periods old, which is approximately $\lceil \log_2 s_2 \rceil$ periods *less* than the delay in the one-tier hierarchy. Resources are allocated between divisions by the center, who uses the cost information $\{p_{j,t-d_2}\}_{j \in \Theta_R}$, which are aggregates of marginal costs from period $t - d_0 - d_1 - d_2$. This cumulative delay is approximately 6 periods greater than the delay of the two-tier hierarchy. As in the hierarchies in Section 4, the decentralized hierarchy has more operations, and the shadow prices ($\hat{p}_{jt}$) at different nodes at a given point in time are different.

Periods $(t-\tau_j-\cdots)$	Calculation	Operations Type	#	Delay
$d_j,\ldots,5$	$\hat{p}_{jt} := \frac{1}{n}\sum_{k\in\Theta_j}\tilde{p}_{kt}$	$+,\times$	s_j	$\lceil\log_2 s_j\rceil+1$
$4,3$	$\Delta^p_{kt} := \alpha_j(\hat{p}_{jt}-\tilde{p}_{kt})$	$-,\times$	$2s_j$	2
$4,3$	$\Delta^x_{kt} := \frac{n_k}{n_j}(x_{jt}-x_{j,t-1})$	$-,\times$	s_j+1	2
$2,1$	$x_{kt} := x_{k,t-1}+\Delta^p_{kt}+\Delta^x_{kt}$	$+$	$2s_j$	2
		Total:	$6s_j+1$	$\lceil\log_2 s_j\rceil+5$

$$\tau_j = 4(h_j-1)$$
$$d_j = \lceil\log_2 s_j\rceil+5$$
$$\tilde{p}_{kt} = \hat{p}_{k,t-d_j-4(h_j-h_k-1)}$$

TABLE 6. Calculations by office $j\in J$ for the period-t allocation in the hierarchical iterative procedure.

References

Baiman, S., Evans, J., and Noel, J. (1987). Optimal contracts with a utility-maximizing auditor. *Journal of Accounting Research*, 25, 217–244.

Baliga, S. and Sjöström, T. (1998). Decentralization and collusion. *Journal of Economic Theory*. Forthcoming.

Baron, D. and Besanko, D. (1984). Regulation, asymmetric information and auditing. *RAND Journal of Economics*, 50, 447–470.

Beggs, A. W. (1995). Queues and hierarchies. Wadham College, Oxford University.

Bernussou, J. and Titli, A. (1982). *Interconnected Dynamical Systems: Stability, Decomposition and Decentralization*. Amsterdam: North-Holland.

Bolton, P. and Dewatripont, M. (1994). The firm as a communication network. *Quarterly Journal of Economics*, 109, 809–839.

Calvo, G. and Wellisz, S. (1980). Technology, enterpreneurs, and firm size. *Quarterly Journal of Economics*, 4, 663–677.

Demski, J. and Sappington, D. (1987). Hierarchical regulatory control. *RAND Journal of Economics*, 18, 77–97.

Dirickx, Y. M. I. and Jennergren, L. P. (1979). *Systems Analysis by Multilevel Methods*. Chichester, England: John Wiley and Sons.

Dudkin, L. M., Rabinovich, I., and Vakhutinsky, I. (1987). *Iterative Aggregation Theory*. New York: Marcel Dekker, Inc.

Friedman, E. J. and Oren, S. S. (1995). The complexity of resource allocation and price mechanisms under bounded rationality. *Economic Theory*, 6, 225–250.

Geanakoplos, J. and Milgrom, P. (1991). A theory of hierarchies based on limited managerial attention. *Journal of the Japanese and International Economies*, 5, 205–225.

Hayek, F. A. v. (1945). The use of knowledge in society. *American Economic Review*, 35, 519–530.

Heal, G. M. (1969). Planning without prices. *Review of Economic Studies*, 36, 347–362.

Hong, L. and Page, S. (1994). Reducing informational costs in endowment mechanisms. *Economic Design*, 1, 103–117.

Klibanoff, P. and Poitevin, M. (1995). A theory of decentralization based on limited commitment. Northwestern University and Université de Montréal.

Laffont, J.-J. and Martimort, D. (1997). Collusion under asymmetric information. *Econometrica*, 65, 875–911.

Laffont, J.-J. and Martimort, D. (1998). Collusion and delegation. *RAND Journal of Economics*, 29, 280–305.

Malone, T. W. and Smith, S. A. (1988). Modeling the performance of organizational structures. *Operations Research*, 36, 421–436.

Marschak, J. and Radner, R. (1972). *Economic Theory of Teams*. New Haven, CT: Yale Univeristy Press.

Marschak, T. and Reichelstein, S. (1995). Communication requirements for individual agents in networks and hierarchies. In J. Ledyard (Ed.), *The Economics of Informational Decentralization: Complexity, Efficiency and Stability*. Boston: Kluwer Academic Publishers.

Marschak, T. and Reichelstein, S. (1998). Network mechanisms, informational efficiency, and hierarchies. *Journal of Economic Theory*, 79, 106–141.

McAfee, R. P. and McMillan, J. (1995). Organizational diseconomies of scale. *Journal of Economics and Management Strategy*, 4, 399–426.

Meagher, K. and Van Zandt, T. (1998). Managerial costs for one-shot decentralized information processing. *Review of Economic Design*, 3. Forthcoming.

Melumad, N., Mookherjee, D., and Reichelstein, S. (1992). A theory of responsibility centers. *Journal of Accounting and Economics*, 15, 445–484.

Melumad, N., Mookherjee, D., and Reichelstein, S. (1995). Hierarchical decentralization of incentive contracts. *RAND Journal of Economics*, 26, 654–672.

Melumad, N., Mookherjee, D., and Reichelstein, S. (1997). Contract complexity, incentives and the value of delegation. *Journal of Economics and Management Strategy*, 6.

Mookherjee, D. and Reichelstein, S. (1995). Incentives and coordination in hierarchies. Boston University and Haas School of Business (UC Berkeley).

Mookherjee, D. and Reichelstein, S. (1997). Budgeting and hierarchical control. *Journal of Accounting Research*, 35, 129–158.

Mount, K. and Reiter, S. (1990). A model of computing with human agents. Discussion Paper No. 890, Center for Mathematical Studies in Economics and Management Science, Northwestern University.

Mount, K. R. and Reiter, S. (1996). A lower bound on computational complexity given by revelation mechanisms. *Economic Theory*, 7, 237–266.

Orbay, H. (1997). Information processing hierarchies. Koç University.

Poitevin, M. (1995). Contract renegotiation and organizational design. Center for Mathematical Studies in Economics and Management Science Discussion Paper No. 1135, Northwestern University.

Qian, Y. (1994). Incentives and loss of control in an optimal hierarchy. *Review of Economic Studies*, 61, 527–544.

Radner, R. (1972a). Normative theories of organizations: An introduction. In C. B. McGuire and R. Radner (Eds.), *Decision and Organization*, chapter 9, pp. 179–188. Amsterdam: North-Holland. Second edition published in 1986 by University of Minnesota Press.

Radner, R. (1972b). Teams. In C. B. McGuire and R. Radner (Eds.), *Decision and Organization*, chapter 10, pp. 189–215. Amsterdam: North-Holland. Second edition published in 1986 by University of Minnesota Press.

Radner, R. (1993). The organization of decentralized information processing. *Econometrica*, 62, 1109–1146.

Radner, R. and Van Zandt, T. (1992). Information processing in firms and returns to scale. *Annales d'Economie et de Statistique*, 25/26, 265–298.

Reichelstein, S. and Reiter, S. (1988). Game forms with minimal message spaces. *Econometrica*, 56, 661–700.

Reiter, S. (1996). Coordination and the structure of firms. Northwestern University.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton, NJ: Princeton University Press.

Van Zandt, T. (1998a). Decentralized information processing in the theory of organizations. In M. Sertel (Ed.), *Economic Design and Behavior*. Proceedings of the XIth World Congress of the International Economic Association, volume IV. London: Macmillan Press Ltd.

Van Zandt, T. (1998b). Real-time decentralized information processing as a model of organizations with boundedly rational agents. *Review of Economic Studies*. Forthcoming.

Van Zandt, T. (1998c). Real-time hierarchical resource allocation with quadratic costs. Princeton University.

Van Zandt, T. (1998d). The scheduling and organization of periodic associative computation: Efficient networks. *Economic Design*, 3, 93–127.

Van Zandt, T. (1998e). Structure and returns to scale of real-time hierarchical resource allocation. Princeton University.

Van Zandt, T. and Radner, R. (1998). Real-time decentralized information processing and returns to scale. Princeton University and New York University.

Williams, S. R. (1986). Realization and Nash implementation: Two aspects of mechanism design. *Econometrica*, 54, 139–151.

Zomaya, A. Y. (Ed.). (1996). *Parallel and Distributed Computing Handbook*. New York: McGraw-Hill.