Real-Time Hierarchical Resource Allocation

Timothy Van Zandt[*]
INSEAD

12 June 2003

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 administrative staff. The control problem is to allocate resources in a changing environment. We consider a class of hierarchical procedures in which information about payoffs 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:

INSEAD
Boulevard de Constance
77305 Fontainebleau CEDEX
France

Voice:	+33 1 6072 4981
Fax:	+33 1 6074 6192
Email:	tvz@econ.insead.edu
Web:	zandtwerk.insead.edu

[*] The comments of Roy Radner and Bob Vanderbei are greatly appreciated. This research was supported in part by grant IRI97-11303 from the National Science Foundation.

Contents

1 Introduction

Understanding the complex structure of such decision making in organizations and markets is important for comparing the variety of institutions through which economic transactions are coordinated. Such decision processes exhibit two forms of decentralization. First, there is decentralized information processing, by which we simply mean that economic decisions are calculated jointly by several or many people. Second, there is decentralized decision making, by which we mean that different decisions are controlled by different people and are made based on different information. Both forms of decentralization lead to many of the contracting and incentive problems that economists usually take as exogenously given.

The purpose of this paper is to construct a model that distinguishes between these two forms of decentralization and that explains some of the advantages and disadvantages of each one. We are particularly interested in providing a new rationale for decentralized decision making.

Our model is specifically of resource allocation procedures that have the hierarchical upward and downward flows of information seen in Figure 1. Examples include the capital budgeting processes of large firms and procedures for allocating resources within large non-market organizations such as governments, firms, and universities. 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 (resource allocations are calculated jointly by the members of the administrative staff) and decentralized decision making (each node makes decisions that constrain the resource allocations and the decisions of different nodes of the hierarchy are calculated using different information).

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

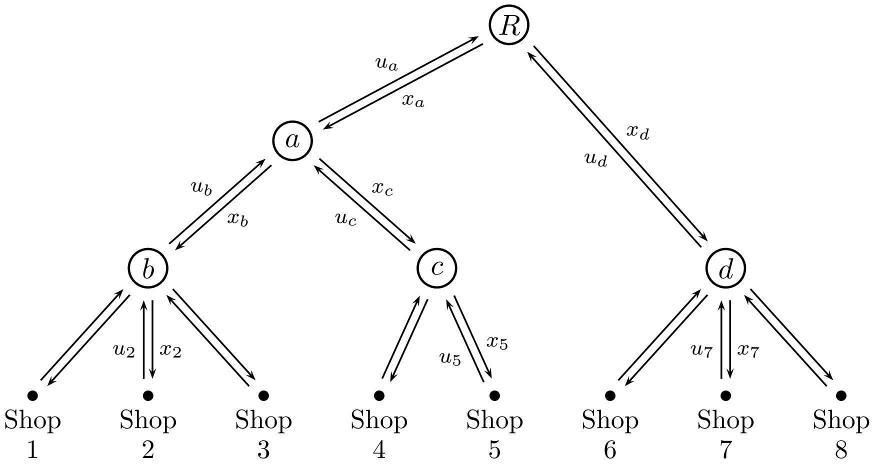

Figure 1. Hierarchical decomposition of the resource allocation problem without externalities. Payoff functions are *aggregated* through an *upward* flow of information. Allocations are *disaggregated* through a *downward* flow of information.

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' payoff information. This is beneficial because each office's information is less aggregated and hence more recent than that of its superior. (The main cost of this decentralization is an increase in the amount of computation.) Stated more broadly, this paper illustrates how decentralization 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 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)). Incentive problems are hard pressed to explain either form, since such decentralization creates rather solves incentive problems. 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 costs. More closely related to this paper is the "batch processing literature" (see Section 3 for references), which 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.

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.

The model in this paper is abstract, which clarifies the main ideas about the advantages and disadvantages of decentralization and demonstrates their generality. Such abstraction is also useful for certain 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

However, it does not permit the statistical assumptions required to quantitatively measure the performance of different hierarchical structures and thereby characterize optimal hierarchies and perform comparative statics. With these goals, a complementary paper (Van Zandt (2003b)) studies a similar model but with very specific assumptions—the shops have quadratic payoff functions whose parameters follow first-order autoregressive processes—which allow such calculations. That paper also takes up several methodological issues that are better suited to the quantitative model, such as a formal treatment of decision procedures and hierarchical structures and an analysis of the distributed statistical inference problem. Van Zandt (2003a, 2003c) 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. These papers also show, for example, that organizations tend to be smaller and more interally decentralized the more quickly the environment is changing.

Reader's guide We begin, in Section 2, by reviewing 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. 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).

2 Hierarchical decomposition of resource allocations

Consider the following one-good resource allocation problem without externalities, framed as a payoff maximization 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

$$\text{(MAX)} \qquad \max_{\{x_i \in X\}_{i=1}^n} \sum_{i=1}^n u_i(x_i) \qquad \text{subj. to: } \sum_{i=1}^n x_i = x_R.$$

The data in the problem are the shops' payoff functions $\{u_1, \dots, u_n\}$ and the amount x_R of the resource. The system is closed in the sense that x_R is fixed.

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 $u_i \colon X \to \mathbb{R}$ could represent profit or negative cost, and it could be a reduced form that subsumes further unmodeled decisions taken within each shop. For example, $u_i(x_i)$ could be shop i's maximum profit when it has capital x_i; $-u_i(x_i)$ could be shop i's minimum cost of producing a quantity x_i of output. 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 u_i. If $x_R = 0$, then the allocations represent net trades. Because payoffs 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 Dirickx and Jennergren (1979) and Bernussou and Titli (1982).

We begin by defining the aggregate of a set of payoff functions. Let $\bar{\mathbb{R}} = \mathbb{R} \cup \{\infty\}$ and let $\bar{\mathcal{U}}$ be the set of functions from X into $\bar{\mathbb{R}}$, the elements of which we call *payoff functions.* The *aggregate* (supremal convolution) of a finite collection $\{u_k \in \bar{\mathcal{U}}\}_{k \in K}$ of payoff functions is the value function of the resource allocation problem with recipients K and payoffs $\{u_k\}_{k \in K}$. That is,[1]

$$u(x) \;\equiv \sup_{\{x_k \in X\}_{k \in K}} \sum_{k \in K} u_k(x_k) \qquad \text{subj. to: } \sum_{k \in K} x_k = x.$$

Denote the aggregate by $\bigoplus_{k \in K} u_k$ or by $u_a \oplus u_b$ if $K = \{a, b\}$. The binary operation $\oplus \colon \bar{\mathcal{U}} \times \bar{\mathcal{U}} \to \bar{\mathcal{U}}$ is associative and commutative and $\bigoplus_{k=1}^m u_k = u_1 \oplus \cdots \oplus u_m$.

Now consider a hierarchy (i.e., a tree) like the one in Figure 1. The leaves are the n shops $\{1, \dots, n\}$. 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.

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

[1] We could define the aggregate of payoff functions $\{u_k \colon X_k \to \mathbb{R}\}_{k \in K}$ with different domains, in which case the domain of the aggregate payoff function is $\sum_{k \in K} X_k$.

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

$$u_j \;=\; \bigoplus\nolimits_{i\in\theta_j} u_i \;=\; \bigoplus\nolimits_{k\in\Theta_j}\Bigl(\bigoplus\nolimits_{i\in\theta_k} u_i\Bigr) \;=\; \bigoplus\nolimits_{k\in\Theta_j} u_k.$$

In Figure 1 this means, for example, that office a can calculate its aggregate payoff function either by directly aggregating the payoff functions $u_1,\ldots,u_5$ of shops $1,\ldots,5$ or by aggregating the aggregate payoff functions u_b and u_c of offices b and c.

Furthermore, we can decompose not only the aggregation of payoff functions, but the disaggregation of resource allocations. Consider the example in Figure 1 and suppose that office a has to allocate an amount x_a of the resource to the shops in division a in order to maximize the division's total payoff. 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 maximize the sum $u_b(x_b)+u_c(x_c)$ of their aggregate payoffs, and then (b) instruct each office b and c to divide its allocation among the shops in its division, in order to maximize the sum of those shops' payoffs.

This property can be stated formally as follows. Let $\{x_i^*\}_{i=1}^n$ be a balanced resource allocation. For $j\in J$, let $x_j^*=\sum_{i\in\theta_j}x_i^*$ be the total resource allocation to the shops below j in the hierarchy. Then $\{x_i^*\}_{i=1}^n$ solves (MAX) if and only if, for each $j\in J$, $\{x_k^*\}_{k\in\Theta_j}$ solves

$$\max_{\{x_k\in\mathbb{R}\}_{k\in\Theta_j}} \sum_{k\in\Theta_j} u_k(x_k) \qquad \text{subj. to: } \sum_{k\in\Theta_j} x_k = x_j^*.$$

This suggests the following hierarchical procedure. Recursively starting at the bottom of the hierarchy, each office calculates its own aggregate payoff function from those of its subordinates, and sends the result to its superior. This stage ends when the root has calculated the overall aggregate payoff function. Then, recursively starting with the root, each office allocates to its subordinates resources received from its superior, in order to maximize the total payoff of its subordinates. The upward flow of payoff 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 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 payoffs—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 (2001), Bolton and Dewatripont (1994), Friedman and Oren (1995), Malone and Smith (1988), Meagher and Van Zandt (1998), Mount and Reiter (1990, 1996), Orbay (2002), Radner (1993), Reiter (1996), and Van Zandt (1998). 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 (1998), 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—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 $f\colon \mathcal{U}^n \times X \to X^n$, where $f(u_1, \ldots, u_n, x_R)$ is the solution to (MAX) and $\mathcal{U} \subset \bar{\mathcal{U}}$ is the set of potential payoff functions. Assume that $\mathcal{U}$ is a set of strictly convex and differentiable payoff functions such that $\mathcal{U}$ is closed under the operation $\oplus$ and such that the resource allocation problem (MAX) has a solution for all $\{u_i \in \mathcal{U}\}_{i=1}^n$ 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 payoff functions) $f_1\colon \mathcal{U}^2 \to \mathcal{U}$, where $f_1(u_a, u_b) = u_a \oplus u_b$;
2. (derivative of a payoff function) $f_2\colon \mathcal{U} \times X \to \mathbb{R}$, where $f_2(u, x) = u'(x)$;
3. (inverse derivative of a payoff function) $f_3\colon \mathcal{U} \times \mathbb{R} \to \mathbb{R}$, where $f_3(u, p) = u'^{-1}(p)$.

These elementary operations are sufficient to compute $f(u_1, \ldots, u_n, x_R)$ as follows.

1. The aggregate payoff function $u_R := u_1 \oplus \cdots \oplus u_n$ can be calculated with $n - 1$ of the operation f_1.
2. The shadow price $p_R := f_2(u_R, x_R)$ 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 payoff equal to the shadow price: $x_i := f_3(u_i, p_R)$. 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 $f(u_1, \ldots, u_n, x_R)$.

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 all raw data and partial results. Therefore, it does not matter 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, where each operation's inputs are as data or outputs of a previously performed operations. As in this paper, it is typically possible to describe computation procedures informally yet clearly. However, Van Zandt (2003b) 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 (2003c), for example, we compare the information processing for different numbers n of shops. Therefore, the aggregation $u_1 \oplus \cdots \oplus u_n$ of all the payoff 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 $f\colon \mathcal{U}^n \times X \to X^n$. 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

[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.

and stylized 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.

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 (1999b, Section 5). We have chosen a model that contains only calculation delays because with "reading delays" there is an implicit communication cost (the managerial wages for the time it takes agents to read messages), which could obscure the fact that decentralized decision making arising in the model 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 (2003b), 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.

[3] Indeed, the operation of aggregating two payoff functions is generally not "elementary" at all; see Section 7.2 for discussion of iterative adjustment procedures that avoid the aggregation of entire functions.

Calculation	Elementary operation	Number of operations	Parallel delay
$u_R := u_1 \oplus \cdots \oplus u_n$	$f_1(u_A, u_B) = u_A \oplus u_B$	$n-1$	$\lceil \log_2 n \rceil$
$p_R := u'_R(x_R)$	$f_2(u, x) = u'(x)$	1	1
$\{x_i := u_i'^{-1}(p_R)\}_{i=1}^n$	$f_3(u, p) = u'^{-1}(p)$	n	1
	Total:	$2n$	$2 + \lceil \log_2 n \rceil$

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

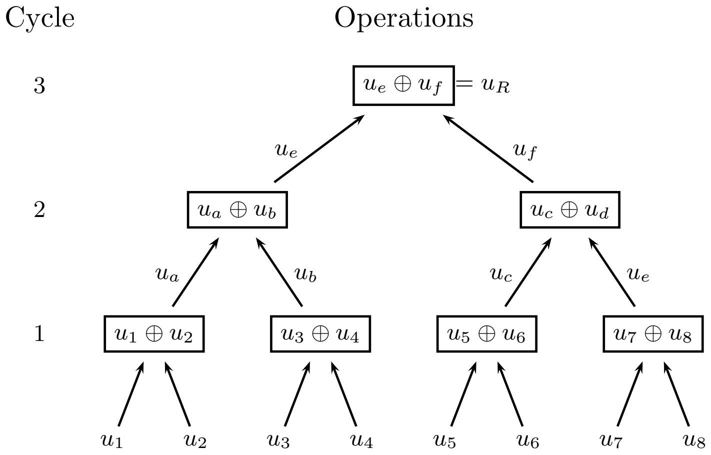

FIGURE 2. Associative computation by a PRAM.

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 $2n$.

Compare this with *decentralized* information processing, in which potentially many agents compute the resource allocation jointly. In the first stage, $u_R := u_1 \oplus \cdots \oplus u_n$ is calculated. The efficient algorithms for associative computation with a PRAM are illustrated in Figure 2. In the first cycle, the payoff functions are divided into pairs, and each pair is assigned to a different agent, who calculates the aggregate of the two payoff functions. In each subsequent cycle, the aggregate payoff 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 u_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 u_R. However, the computation of u_R still requires $n-1$ operations.

The next step is to compute $p_R := f_2(u_R, x_R)$, which one agent does in one cycle. Finally, the n operations $\{x_i := f_3(u_i, p_R)\}_{i=1}^n$ 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, the aggregation of payoff functions in Figure 1 is similar to the decentralized aggregation of payoff functions in our batch processing model. However, the disaggregation of resource allocations in Figure 1 has no analog.

Consider first the aggregation of payoff 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 payoff 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 $u_a := u_b \oplus u_c$ in the third cycle and agent R computes $u_R := u_a \oplus u_d$ 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 u_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, there is a recursive disaggregation of resource allocations. Starting with the root, each agent j receives the allocation x_j for its division, calculates its shadow price $p_j := u'_j(x_j)$, and then allocates resources to each subordinate $k \in \Theta_j$ by setting x_k so that $u'_k(x_k) = p_j$. With such a procedure, the shadow price is the same at every node and the only useful operations are the calculation of the overall shadow price, $p_R := u'_R(x_R)$, and of the individual shops' allocations, $x_i := u_i'^{-1}(p_R)$. 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 cost (as is done, for example, in Radner (1993) and Van Zandt (1998)). 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 cost and the "cost" of delay. Administrative cost is 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 cost 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 payoff functions $\{u_{1t}, \dots, u_{nt}\}$ are realized and observed and a deterministic quantity x_{Rt} of the resource must be allocated. That is, the payoff in period t when the allocation of x_{Rt} is $\{x_{1t}, \dots, x_{nt}\}$ equals $\sum_{i=1}^{n} u_{it}(x_{it})$. 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 payoff functions.

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 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.

We measure the net performance in each period by the expected payoff 0of the decision rule minus the the administrative cost of the computation procedure, and we call it the *profit*. In order to determine the expected payoff of a decision rule, we need assumptions about the stochastic process $\{u_{1t}, \dots, u_{nt}\}_{t=-\infty}^{\infty}$. 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 a lower expected payoff than those that use recent information.

In this real-time setting, the following stationary procedures resemble batch processing and compute the allocation for each period from data of homogeneous lags. The resource allocation for each period t is calculated, in the manner described in Section 3, from the payoff functions $\{u_{1,t-d}, \dots, u_{n,t-d}\}$ collected in period $t-d$.[4,5] The lag d is the delay in

[4] With specific statistical assumptions, as in Van Zandt (2003b), we could allow the decision rule to take into account the expected change in the payoff functions between periods $t-d$ and period t.

[5] Since the shadow price is calculated from $\{u_{i,t-d_1}\}_{i=1}^{n}$, the final step in which the marginal payoff is set equal to the shadow price must use these same payoff functions (rather than more recent ones) in order to balance the allocation.

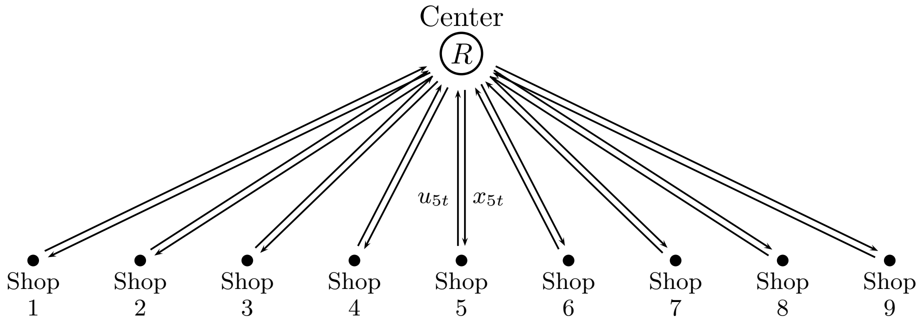

FIGURE 3. A two-tier centralized hierarchy.

performing these computations, which is given in Table 1. With serial processing (one agent), the allocation in each period is calculated from the payoff functions from $2n$ periods ago, whereas with decentralized computation, each allocation is calculated from the payoff functions from $2 + \lceil \log_2 n \rceil$ periods ago. Such decentralization leads to lower delay and hence more recent information and a higher payoff.

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 payoff functions from the shops and sends the current allocation to the shops. In period t, the organization is busy computing the allocations for periods $t+1, \ldots, t+d$. 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. Such a hierarchy is shown in Figure 4 for $n = 9$ and $s_1 = s_2 = 3$.

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 payoff information the aggregate payoff functions calculated by the tier-1

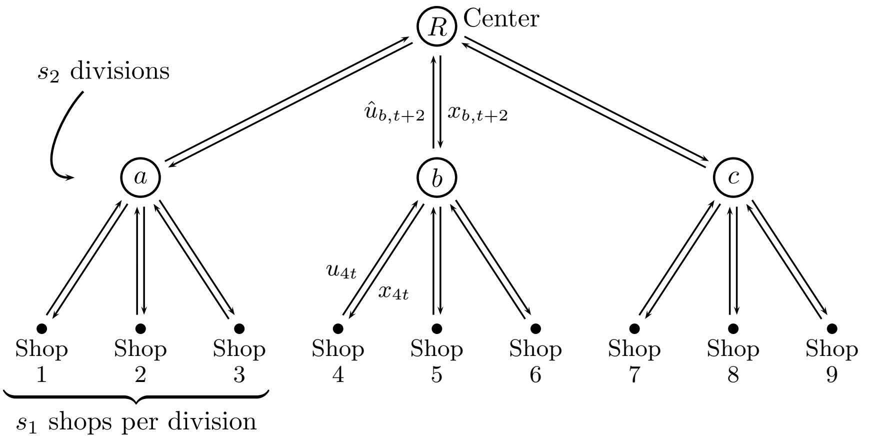

FIGURE 4. Information flow at the beginning of period t in a three-tier hierarchy.

	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	$\{u_{i,t-d_1}\}_{i \in \theta_j}$, x_{jt}	$\{\hat{u}_{j,t-d_2}\}_{j \in \Theta_R}$, x_{Rt}
Aggregation	$\hat{u}_{jt} := \bigoplus_{i \in \theta_j} u_{i,t-d_1}$	$\hat{u}_{Rt} := \bigoplus_{j \in \Theta_R} \hat{u}_{j,t-d_2}$
Shadow price	$p_{jt} := \hat{u}'_{jt}(x_{jt})$	$p_{Rt} := \hat{u}'_{Rt}(x_{Rt})$
Allocation	$x_{it} := u'^{\,-1}_{i,t-d_1}(p_{jt})$	$x_{jt} := \hat{u}'^{\,-1}_{j,t-d_2}(p_{Rt})$

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

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

As information is recursively aggregated and passed up the hierarchy, it gets older. As one might guess, the cumulative lag of the raw data that goes into the center's decisions equals $d_1 + d_2$. However, the timing of the calculations has two complications that cancel each other in the determination of the cumulative lag. In Table 2, we see that division j needs to be informed of its period-t allocation x_{jt} by the beginning of period $t-2$ (in order to have time to disaggregate it), which is also when it finishes calculating its aggregate payoff function $\hat{u}_{jt}$. Hence, the center has to start calculating the period-t allocation in period $t - d_2 - 2$, but the aggregate data from offices in tier 1 available that period are $\hat{u}_{j,t-d_2}$. Thus, the aggregate payoff function $\hat{u}_{Rt}$ that the center calculates in order to allocate the resource for period t is

$$\hat{u}_{Rt} \;=\; \bigoplus\nolimits_{j \in \Theta_R} \hat{u}_{j,t-d_2} \;=\; \bigoplus\nolimits_{j \in \Theta_R} \bigoplus\nolimits_{i \in \theta_j} u_{i,t-d_1-t_2} \;=\; \bigoplus\nolimits_{i=1}^{n} u_{i,t-d_1-d_2} \;.$$

The upward flow of information and downward flow of allocations that take place each period are shown in Figure 4.

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.

It may appear that one difference between the two-tier and the three-tier hierarchies is that, in the latter, the aggregation of payoff 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 payoff functions is always maximally decentralized within each node. This means that the hierarchical decomposition of the aggregation of payoff 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 payoff function $\hat{u}_{Rt}$ that it uses to allocate resources for period t. Furthermore, $\hat{u}_{Rt} = \bigoplus_{i=1}^{n} u_{i,t-d_1-d_2}$. Hence, the aggregate payoff 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 payoff function $\hat{u}_{Rt}$ 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.

> When a division receives x_{jt} from the center in period $t-2$, it does not have to allocate x_{jt} using the information $\hat{u}_{j,t-d_2}$ that the center used to compute x_{jt}. Instead, it uses its most recently calculated aggregate payoff function $\hat{u}_{jt}$. 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.

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).

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). Van Zandt (2003b) generalizes 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.[6] 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

[6] 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.

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,[7] 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 is related to decentralized decision making. This claim is considered more formally in Van Zandt (2003b); 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 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?

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

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 $\{u_{i,t-d}\}_{i=1}^{n}$. In the three-tier hierarchy, the allocations at time t to the shops in division j are a function of the data $\{u_{i,t-d_1-d_2}\}_{i=1}^{n}$ that the center implicitly uses to compute x_{jt} and of the data $\{u_{i,t-d_1}\}_{i\in\theta_j}$ 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.[8]

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 or nonhierarchical procedure that more closely resemble market mechanisms may be more decentralized.

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).

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 (1999a) 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

[8] 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.

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. (Section 7.3 mentions papers that take decentralization as given and consider its consequences for incentives.) Nevertheless, there have been a few incentives-based justifications for decentralization, which we now describe.

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 (1997), 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.

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. We did not 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 (2003b) 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, managers 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 (2003b), as explained there.

Radner and Van Zandt (1992) and Van Zandt and Radner (2001), 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 (1999a, Section 2.1), 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 payoffs 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 (2003b, 2003a, 2003c), 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 payoff 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 payoff functions, as in Van Zandt (2003b)) (rather than calculating resource allocations that are optimal given old payoff 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 maximizing collective profits; 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 maximizes the sum of their payoffs. 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 payoffs 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 $\oplus$ 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 payoff information flows up the hierarchy and resource allocations flow down. The analog in an iterative procedure is that local payoff information, in the form of marginal payoffs (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 payoffs (average shadow prices) of the shops inferior to the offices, and this information is different for different nodes of the hierarchy.

Defining multitier real-time versions of the more classical price–quantity Walrasian procedure in Friedman and Oren (1995) poses several difficulties. 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 (1997, 2001)). 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 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 payoff 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.

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*, 83, 196–232.

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

Beggs, A. W. (2001). Queues and hierarchies. *Review of Economic Studies*, 68, 297–322.

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.

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, 329–345.

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

Mookherjee, D. and Reichelstein, S. (2001). Incentives and coordination in hierarchies. *Advances in Theoretical Economics*, 1.

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. (2002). Information processing hierarchies. *Journal of Economic Theory*, 105, 370–407.

Poitevin, M. (1997). Contract renegotiation and organizational design. Mimeo, Université de Montréal.

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.

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

Van Zandt, T. (1999a). Decentralized information processing in the theory of organizations. In M. Sertel (Ed.), *Contemporary Economic Issues, Vol. 4: Economic Design and Behavior*, chapter 7. London: Macmillan Press Ltd.

Van Zandt, T. (1999b). Real-time decentralized information processing as a model of organizations with boundedly rational agents. *Review of Economic Studies*, 66, 633–658.

Van Zandt, T. (2003a). Balancedness of real-time hierarchical resource allocation. INSEAD.

Van Zandt, T. (2003b). Real-time hierarchical resource allocation with quadratic costs. INSEAD.

Van Zandt, T. (2003c). Structure and returns to scale of real-time hierarchical resource allocation. INSEAD.

Van Zandt, T. and Radner, R. (2001). Real-time decentralized information processing and returns to scale. *Economic Theory*, 17, 497–544.

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.