



**Matching Allocation  
Problems with Endogenous  
Information Acquisition**

Sophie Bade





# **Matching Allocation Problems with Endogenous Information Acquisition**

Sophie Bade

May 2011

# Matching Allocation Problems with Endogenous Information Acquisition

SOPHIE BADE<sup>\*†</sup>

May 12, 2011

## Abstract

The paper introduces the assumption of costly information acquisition to the theory of mechanism design for matching allocation problems. It is shown that the assumption of endogenous information acquisition greatly changes some of the cherished results in that theory: in particular, the first-best might not be implementable. Moreover, it might not even be possible to implement the second-best through trade. In addition, the paper highlights the use of randomness in setting incentives for efficient learning. The trade-offs among simultaneous and sequential learning and among efficient learning and efficient allocations are discussed.

KEYWORDS: Serial Dictatorship, House Allocation Problems, Endogenous Information. *JEL Classification Numbers*: C78.

---

<sup>\*</sup>Max Planck Institute for Research on Collective Goods, Kurt-Schumacher-Str. 10, D-53113 Bonn, Germany.

<sup>†</sup>Thanks to Anne-Katrin Roesler for excellent research assistance. Thanks to Felix Bierbrauer, Christoph Engel, Guillaume Haeringer, Martin Hellwig, Andreas Nicklisch, Timofiy Mylovanov, Ludovic Renou, and the seminar participants at Centro de Modelación Matemática in Santiago de Chile, at the University of California in Berkeley, at the University of Cologne, and at the Universidad Autónoma de Barcelona for their comments.

# 1 Introduction

The central question of allocation theory is how goods and services should be assigned to people. The answer depends on people's valuations for these goods and services. If people know these valuations, the only question is whether this information can be used in an incentive-compatible way. The subfield of allocation theory that is concerned with the design of mechanisms that match indivisible goods to agents has been particularly successful in that respect. For such allocation problems, there is a large class of efficient and incentive-compatible matching mechanisms. However, in many cases, the requisite information cannot be taken as given, but must be acquired at a cost. Whether kidneys, school slots or medical residency programmes need to be matched to agents, it always holds that agents spend time and money to learn about the available options before announcing their choices to the designer. This article is concerned with the design of matching mechanisms when the agents' information is endogenously acquired.

Matching mechanisms specify rules according to which agent choices are mapped to matchings of objects to agents. In the standard case of agents knowing their own preferences, mechanisms implement social choice functions that map profiles of preferences to allocations via the equilibrium behavior of agents. When information acquisition is costly, agents face two kinds of strategic choices. They have to decide on their information acquisition, as well as on the actions to take in the mechanism. Just as in the case with known preferences, mechanisms incentivize the behavior of agents for their *ex post* preferences. However, in the case of costly information acquisition, these *ex post* preferences are - to some extent - endogenous to the agents' behavior, since they depend on the agents' learning choices. Mechanisms do set the incentives for learning choices, as the value an agent assigns to some particular piece of information depends on the usefulness of that information in the mechanism.

In a nutshell: in an environment of endogenous learning, mechanisms implement the learning behavior of agents, as well as social choice functions for *ex post* preferences. Given that these purposes potentially conflict not all *ex-ante* Pareto optima are implementable. The trade-off between the efficient

elicitation of information and the efficient usage of this information has garnered interest in the nascent literature on mechanism design with endogenous information acquisition<sup>1</sup>: Gerardi and Yariv (2008) illustrate this trade-off by showing that welfare-optimal voting rules might not efficiently use all available information, since such a voting rule not only serves to aggregate information, but also to elicit the acquisition of this information. Bergemann and Valimaki (2007) discuss this same trade-off within a framework of auctions.

I show that first-best learning in matching allocation problems is generally sequential. This stands in contrast with the mechanisms that are used in practice to match kidneys or school slots. These mechanisms typically require agents to submit their preferences simultaneously. This suggests that sequential forms of these mechanisms might lead to welfare improvements. The sequentiality of optimal information acquisition has also been featured in the literature on mechanism design with endogenous information acquisition. Gershkov and Szentes (2009) as well as Smorodinsky and Tennenholtz (2006) present voting models in which the voters' optimal acquisition of information is sequential. Similarly, for auctions, Compte and Jehiel (2007) find that ascending price auctions dominate sealed bid auctions in terms of expected welfare and, for a sufficiently large set of bidders, in terms of expected seller revenue.

Trade plays a major role in allocation theory: in matching allocation problems with exogenous information, any one of the plethora of "trading mechanisms" can be used to implement any Pareto-optimal allocation (see for instance Papai (2000)). The result that ex-ante Pareto optima are not always implementable with endogenous information, of course, prevents us from obtaining such a far-reaching result for the present framework. However, one might ask whether the second-best (namely the ex-ante Pareto optima among the implementable set) are implementable through trade. The answer is negative. I provide cases of matching allocation problems, in which some mechanisms yield strictly higher ex ante welfare than the maximal ex-ante welfare that is implementable through trade. The designer can generally improve welfare by retaining more control over the available options than

---

<sup>1</sup>a review of this literature can be found in Bergemann and Valimaki (2007)

any trading mechanism would permit.

There are some similarities and differences between the present paper and the existing literature on endogenous information acquisition in markets (see Grossman and Stiglitz (1980), Verrecchia (1982), and more recently Dang (2008)).<sup>2</sup> One of the main findings of that literature is that the first-best outcome is generally not achievable through trade in the presence of endogenous information acquisition. Here I show that even the second-best might not be implementable through trade in the context of matching problems with endogenous information acquisition. Different externalities generate the different sub-optimality: in the literature on markets it is generally assumed that agents can acquire information on some common value. So there is an informational externality. Very differently in my model, the different agents' values of the same house are independent draws. So there is no informational externality. A different kind of externality is present in my model: the fact that mechanisms need to determine matchings implies an interdependency of the assignments to the different agents. This, in turn, means that if some house  $h^*$  is offered to some agent to give him the incentive to acquire a signal on yet another house, the same house cannot be offered to a different agent at the same time.

There is another feature of optimal matching mechanisms for housing problems with endogenous information acquisition that I wish to highlight: their use of randomness. On the one hand, the observation that randomness aids implementation could count as a stylized fact of the theory of mechanism design. On the other hand, there is a perfectly non-random mechanism that can be used to implement any Pareto optimum in a matching allocation problem with exogenous information: serial dictatorship. In a serial dicta-

---

<sup>2</sup>One of the basic tenets of this literature is that the equilibrium price does not perfectly aggregate the available information when agents can endogenously decide whether to acquire costly information about the values of the traded assets. If prices perfectly aggregated this information, no agent would have any incentive to acquire information. For such incentives to exist, the agents who choose to obtain (costly) information must end up better informed than those who choose not to acquire information. As a consequence, markets with endogenous information acquisition are inefficient. These results are driven by the assumption that information concerns the *common* value of some asset; conversely, the present paper concerns purely idiosyncratic values.

torship, a first agent (the first dictator) gets to choose one house out of the grand set, a next agent (the second dictator) gets to choose one house out of the remainder. This process goes on until no houses are left. Observe that any action an agent takes in this mechanism has a non-random outcome.

Once again, the case of matching allocation problems with endogenous information acquisition differs starkly from the standard case: I provide examples of housing problems in which the second-best cannot be implemented through mechanisms in which each action corresponds to the deterministic appropriation of some house. Instead, there have to be some actions with uncertain outcomes. Randomness is a valuable tool to incentivize learning. Moreover, randomness is useful to attenuate the clash between the objectives of efficient learning and efficient allocations. Since randomness can be generated via simultaneous learning, it might well be that agents need to learn simultaneously according to the second-best mechanism - even if the first-best strategy always prescribes sequential learning.

I frame all these questions in a very simple environment of “house allocation problems”, in which some objects, henceforth called houses, need to be (bijectively) matched to some agents. The model deviates as parsimoniously as possible from the model of exogenous information inasmuch as as that agents are able to learn the value of at most one house each. To avoid the issue of informational externalities, the values of the houses are assumed to be independently distributed. I assume, moreover, that agents are ex-ante identical in the sense that they all assign the same expected values to houses that they have not investigated. This assumption of ex-ante identity makes it reasonable to focus on mechanisms that maximize ex-ante welfare (as opposed to ex-ante Pareto optimality). In the given model, the maximization of ex-ante welfare corresponds to the maximization of expected individual utility if each agent has an equal chance to obtain any role within the mechanism. Since ex-ante welfare optima are also ex-ante Pareto optima, the results on welfare optima not being implementable directly imply that some ex-ante Pareto optima are not implementable.

The paper is structured as follows: first, I discuss the main concepts and findings at the hand of an introductory example (Section 2). In Section 3, I formally define the model and the important concepts of the paper. I go on to

characterize first-best behavior within the model (Section 4). In the following Section 5, I show that some learning and allocative behavior in a matching allocation problem is implementable, if and only if it is implementable by a suitably defined “sequential revelation mechanism”. Building on the results from the prior two sections, Section 6 provides a sufficient and necessary condition for the implementability of the first-best. In Section 7, I show that the second-best need not be implementable through trade. Section 8 is devoted to the trade-offs between efficient learning and efficient allocations and between the sequential and the simultaneous elicitation of preferences. Section 9 concludes.

## 2 An Introductory Example

### 2.1 The Housing Problem

Consider a housing problem with three houses  $h_1$ ,  $h_2$  and  $h_3$  and three agents. Assume that the agents’ true valuations of  $h_1$  and  $h_2$  are drawn from binary distributions  $(p_1, A_1, a_1) = (\frac{3}{4}, 8, 0)$  and  $(p_2, A_2, a_2) = (\frac{1}{2}, 3, -2)$  (with the interpretation that  $h_1$  is of high value  $A_1 = 8$  with probability  $p_1 = \frac{3}{4}$ , and of low value  $a_1 = 0$  otherwise, ditto for  $h_2$ ). These draws are independent across houses and agents. The value of house  $h_3$  is known to be  $\alpha_3 = -1$ . Consequently, each agent has the same a priori ranking  $h_1 \succ h_2 \succ h_3$  over the three houses. Agents 1 and 2 can each investigate one house of their choosing. Agent 3 evaluates  $h_1$ ,  $h_2$  and  $h_3$  by their expected values  $\alpha_1 = 6$ ,  $\alpha_2 = .5$  and  $\alpha_3 = -1$  in turn.

In terms of substance, these assumptions can be interpreted as follows. The agents agree on an a priori ranking that depends on publicly known aspects of the houses, such as whether a house is large or small, whether it lies right by the highway or not, etc. A priori (before any investigations have taken place) all agents rank the houses by their agreed expected values of  $\alpha_i = p_i A_i + (1 - p_i) a_i$ . The independence assumption implies that agents value the initially unknown features of houses in a truly idiosyncratic manner: the walls of a house might, for example, be painted in a shade of blue that to some appears “fresh”; they would assign value  $A_i$  to the house, whereas

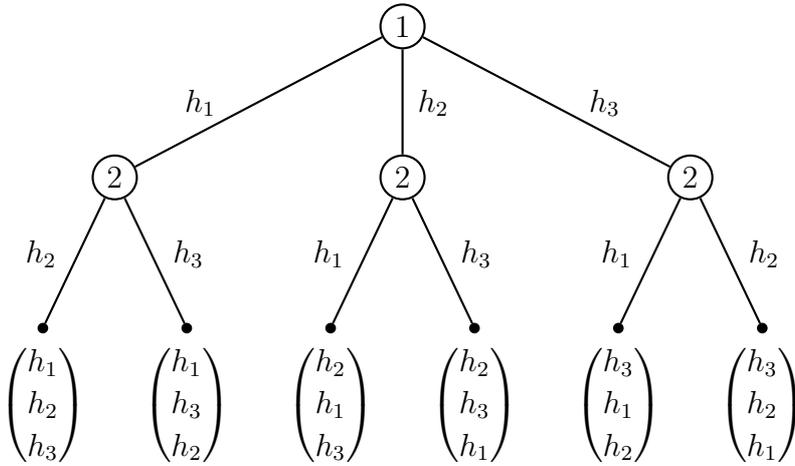


Figure 1: Serial Dictatorship

to others the same shade would just appear “icy”.

## 2.2 Serial Dictatorship

To get a grasp on the effects that mechanisms may have on information acquisition *as well as* on allocations, consider serial dictatorship. In a serial dictatorship with three houses, some agent (the first dictator) gets to choose a house out of the set  $\{h_1, h_2, h_3\}$ , then another agent (the second dictator) gets to pick a house out of the remainder. Figure 1 graphically represents serial dictatorship. The tree in Figure 1 is considered a *rule-tree*, since it summarizes the rules set by the mechanism designer. The nodes are labeled with the agents, the vertices represent actions. The outcome vectors represent allocations, where the top (bottom) element represents agent 1’s (3’s) assignment.

To comprehend the game that is being played under a given set of rules, we also need to consider the learning decisions of agents. To this end, I assume that, right before choosing, agents 1 and 2 may investigate any house. The information whether and which house an agent investigates is private, just like the outcome of that investigation. The induced game tree is large, Figure 2 provides a partial sketch of it. Any node in this tree that is not explicitly labeled with payoffs is to be considered a non-terminal node. Note

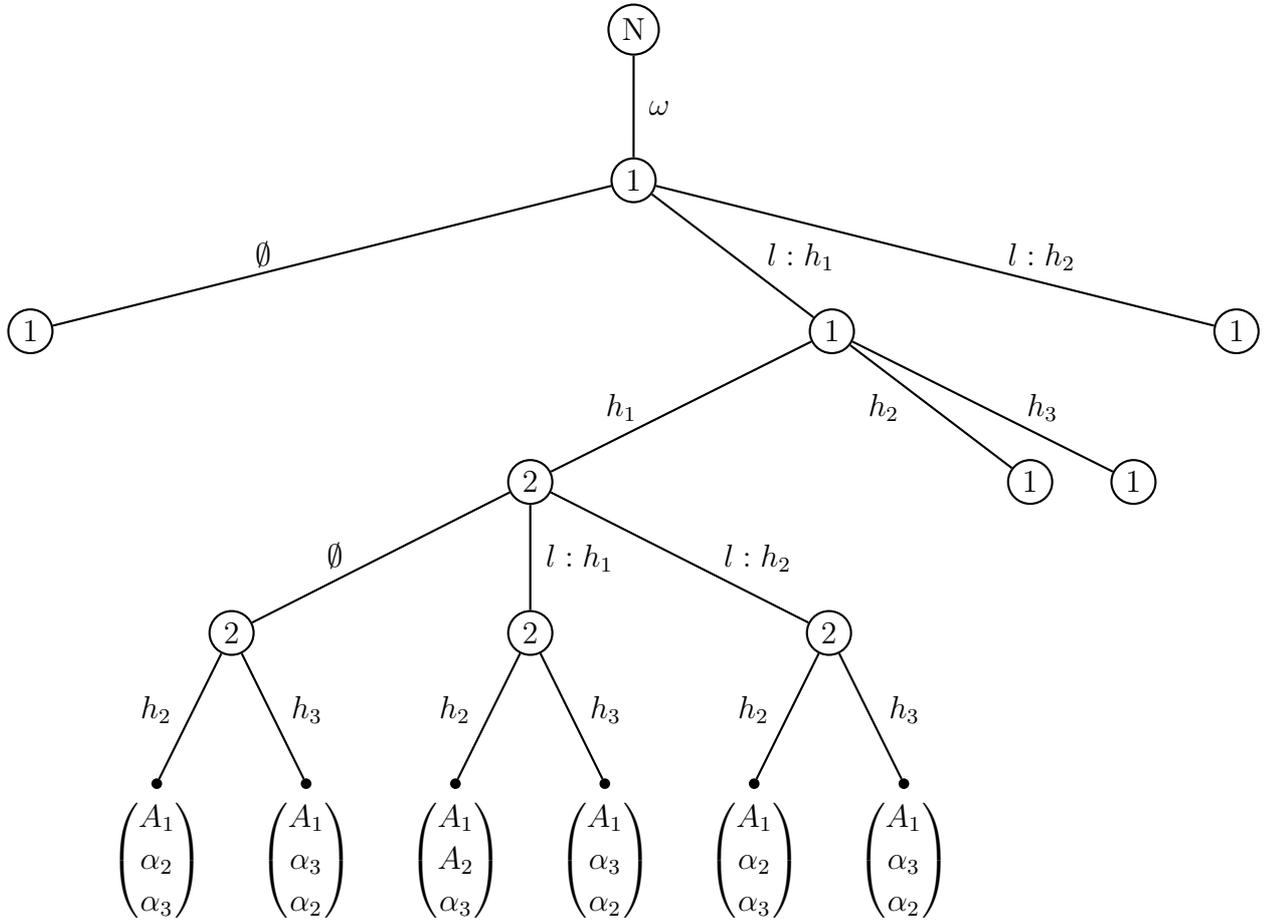


Figure 2: Sketched Game Tree: Serial Dictatorship

that the game starts with a move of nature, in which the agents' valuations of the houses are determined. I only sketched the subtree following the draw of the state  $\omega$ , in which every agent values every house highly. The initial node  $N$  of the (full) tree has a branch for every possible profile of the agents' preferences.

Any one of nature's moves is followed by a decision node for agent 1, in which he has to choose whether and which house to investigate (actions:  $l : h_1$ ,  $l : h_2$  and  $\emptyset$ ). All these "learning" nodes of agent 1 belong to the same information set. Any one of these learning choices is followed by another node for agent 1; in each of these, agent 1 can choose one out of the 3 houses.

I chose to only sketch the subtree according to which agent 1 investigates house  $h_1$  and chooses that same house. Once agent 1 has chosen a house, it is up to agent 2 to decide whether and which house to investigate. Each of these possible choices of an investigation is followed by the choice of a house that is still available ( $h_2$  or  $h_3$  in the subtree I chose to draw).

The payoffs reflect the knowledge of agents right before they move into their respective houses. Consider the case in which agent 2 is assigned  $h_2$ . If he did investigate  $h_2$ , he obtains  $A_2$  for the given state  $\omega$ ; if not, he just obtains  $\alpha_2$ . To complete the tree information-sets would have to be drawn as well. They are such that learning is private. When agent 2 decides whether and which house to investigate, he knows which house agent 1 has chosen; he does, however, not know anything about agent 1's preceding investigation.

To find the unique perfect Bayesian equilibrium, observe that there is only one case in which  $h_1$  is not agent  $j$ 's most preferred house: agent  $j$  did investigate  $h_1$  and found it to be of low value. In this case, agent  $j$  ranks  $h_2$  at the top. It is therefore optimal for the first dictator to investigate  $h_1$ , to keep it if he finds it of high value and to keep  $h_2$  otherwise. Consequently, there are only two choice sets on the equilibrium path for the second dictator: if the first dictator has chosen to keep  $h_1$ , the second gets to choose from  $\{h_2, h_3\}$ , otherwise the second dictator gets to choose from  $\{h_1, h_3\}$ . In the latter case, the second dictator will choose  $h_1$  without any investigation, since he prefers  $h_1$  to  $h_3$ , no matter whether the value of  $h_1$  is high or low (since  $a_1 = 0 > -1 = \alpha_3$ ). In the first case, it is best for the second dictator to learn his valuation of  $h_2$  and to choose it, if and only if it is of high value (since  $A_2 = 3 > \alpha_3 = -1 > a_2 = -2$ ).

It is useful to separate the agents' learning choices analytically from the choices that matter for the allocation. This allows me to consider separately efficient allocations for given ex post preferences and efficient learning. It also facilitates the use backwards induction of determine first-best behavior.

The learning choices are represented by *learning trees*, which specify follow-up investigations for any history of outcomes of preceding investigations. In the case under discussion, the learning tree starts with agent 1's investigation of  $h_1$ . Learning continues with agent 2's investigation of  $h_2$ , if and only if agent 1 has found  $h_1$  to be of high value; otherwise agent 2

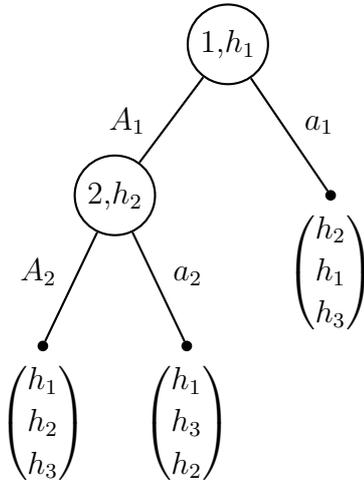


Figure 3: Learning Tree, Serial Dictatorship

will not investigate any house. The choices that matter for the allocation of the houses are summarized by an allocation function which maps any (feasible) a posteriori preference profiles to an allocation. Figure 3 illustrates the learning tree and allocation function implemented by serial dictatorship: nodes correspond to investigations, branches to values, and outcomes to the allocations prescribed for the different a posteriori preference profiles.

### 2.3 First-Best Learning and Allocations

Serial dictatorship is but one example of a mechanism. In this subsection, I investigate the question whether there is some learning behavior and allocation functions that yield higher ex ante welfare than the learning behavior and allocation function implemented by serial dictatorship. To determine first-best learning behavior and allocation functions, abstract away from any incentive constraints. Start by fixing some profile of a posteriori preferences. If possible,  $h_1$  should be assigned to someone who values it highly. If no one values  $h_1$  highly, it should be assigned to someone who did not investigate it. The same holds for  $h_2$ . Since agent 3 cannot investigate any house, such an allocation is always feasible.

This description of efficient allocations has two immediate implications

for first-best learning: firstly, if an agent found some house to be of high value, it makes no sense to let the other agent investigate the same house. Secondly, both options to investigate a house should always be used. These two preliminary observations leave us with just three candidates for the first-best learning tree. Either one agent investigates  $h_1$  and the other investigates  $h_2$ . Or one agent investigates  $h_1$  and the next agent investigates  $h_1$ , if and only if the first found  $h_1$  to be of low value. Exchanging  $h_1$  and  $h_2$ , one obtains the third alternative. The welfare of three learning trees is  $O_1 + O_2 + B$ ,  $O_1 + (1 - p_1)O_1 + p_1O_2 + B$ , and  $O_2 + (1 - p_2)O_2 + p_2O_1 + B$ , with  $B: \alpha_1 + \alpha_2 + \alpha_3$  and  $O_i: p_i(A_i - \alpha_i)$  for all  $i$ . The term  $B$  can be interpreted as a base value of welfare which arises when randomly assigning houses. Next,  $O_i$  is called *option value* of  $h_i$ . To gain some intuition for this statistic, consider a hypothetical choice problem in which an agent gets to choose one of two houses with identically and independently distributed values. The option value of a house is the agent's maximal willingness to pay for the right to learn the value of one of these two houses. Intuitively, the option value summarizes the value of learning. In this particular case,  $p_1(A_1 - \alpha_1) = 1.5$  and  $p_2(A_2 - \alpha_2) = 1.25$  hold, and therefore the optimal learning tree prescribes that agent 1 investigates house  $h_1$  and agent 2 conditions his investigation on the outcome of that first investigation. This observation is generalized in Lemma 1 in Section 4, which shows that according to the first-best learning tree and allocation function houses should be investigated in order of their option value.

## 2.4 Revelation Principle

In the standard case, without endogenous learning, the search for an optimal mechanism is greatly simplified by the revelation principle, which states that any social choice function that can be implemented by some mechanism can also be implemented by a direct revelation mechanism. How does this principle translate to the environment with endogenous learning? Can any implementable learning tree and allocation function also be implemented by a mechanism in which the designer simultaneously and truthfully elicits the agents' types? The preceding discussion of the first-best learning tree imme-

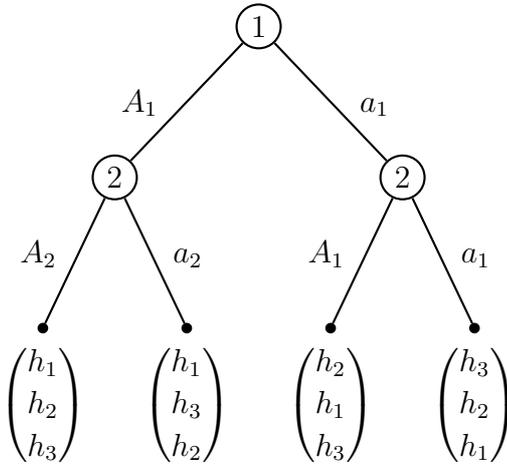


Figure 4: Rule Tree of Sequential Revelation Mechanism

diately yields a negative answer to this question. According to the first best learning tree, agent 2 conditions his investigation of a house on the outcome of the first investigation. Consequently, there exists no mechanism which, on the one hand, implements the first best learning tree and, on the other hand, has simultaneous announcements of preferences. In Section 5, I show a dynamic version of the revelation principle, according to which a learning tree and allocation function are implementable if and only if they are implementable by a mechanism according to which the designer truthfully elicits the agents' types in the proper sequence. The following subsection illustrates the principle of sequential truthful revelation.

## 2.5 Implementation of the First-best

The following mechanism implements the first-best learning tree and allocation function by *sequential truthful implementation*: if agent 1 declares  $A_1$ , he gets  $h_1$ ; if not, he is wait-listed. In the first case, agent 2 gets house  $h_2$ , if he declares  $A_2$ , and  $h_3$  otherwise. In the second case, the allocation is  $(h_2, h_1, h_3)$  if agent 2 declares  $A_1$ , and  $(h_3, h_2, h_1)$  otherwise. The mechanism is illustrated by the rule-tree in Figure 4.

To see that truthful revelation is equilibrium behavior consider the decision problems of the two agents. If agent 1 did announce  $A_1$ , agent 2's announcement of  $A_2$  or  $a_2$  effectively corresponds to a choice between  $h_2$  and  $h_3$ . I already observed in the discussion of serial dictatorship (Section 2.2) that an agent who faces this choice maximizes utility by investigating  $h_2$  and keeping it, if and only if he finds it to be of high value. Similarly, if agent 1 has announced  $a_1$ , the choice of agent 2 is effectively one between  $h_1$  and  $h_2$ , which implies that investigating  $h_1$  and telling the truth is a best reply for agent 2 in this node.

Now let us consider agent 1's decision whether to announce  $A_1$  or  $a_1$ . If he announces  $A_1$ , he is being assigned  $h_1$ ; if he announces  $a_1$ , his assignment depends on the decision of agent 2. If agent 2 announces  $A_1$ , agent 1 is assigned  $h_2$ ; otherwise he is assigned  $h_3$ . But we just showed that agent 2 best responds by announcing  $A_1$ , if and only if he finds  $h_1$  to be of high value, which in turn happens with a probability of  $\frac{3}{4}$ . Given the choice between  $h_1$  and the lottery according to which he obtains  $h_2$  with probability  $\frac{3}{4}$  and  $h_3$  otherwise, an investigation of  $h_1$  is strictly preferable to no investigation, as the expected value of the lottery lies strictly between the two possible values of  $h_1$ . Investigating  $h_1$  is preferable to investigating  $h_2$ , since knowing the value of  $h_2$  has no impact on the agent's preference among  $h_1$  and  $h_2$  ( $h_1$  is preferable to the lottery no matter whether the value of  $h_2$  is high or low).

Observe that the learning tree implemented by the mechanism is indeed the first-best learning tree. To see that the mechanism also implements a first-best allocation function, observe that any agent who claims to have found a house to be of high value obtains that house, and no agent who claims to have found a house to be of low value obtains this house.

Note that agent 1 faces some randomness in the sequential revelation mechanism: If he claims that he found  $h_1$  to be of low value, his assignment depends on the action of agent 2. In equilibrium agent 2 investigates  $h_1$  and truthfully declares the result of his investigation. Consequently, agent 1 faces some randomness on the equilibrium path. I argue in Section 7 that such randomness is actually necessary for the implementation of the ex ante welfare optimum in the present example. Also note that the sequential revelation mechanism is not a trading mechanism. In Section 7 I argue

that there is no trading mechanism which would also implement the ex-ante welfare optimum in the present example.

## 3 The Model

### 3.1 Agents, Houses, Values

There are finite sets of agents  $N: = \{1, \dots, n\}$  and of houses  $H: = \{h_1, \dots, h_n\}$ . Agent  $j$  values house  $h_i$  at  $\omega_i^j$ . The goal of the designer is to maximize welfare by matching houses to agents; formally, the objective is to find a bijection  $\mu: H \rightarrow N$  such that  $\sum_i \omega_i^{\mu(i)}$  is maximized. This problem is easily solved, if all values  $\omega_i^j$  are known to the designer. If this is not the case, the question is whether the designer has any means of learning the values  $\omega_i^j$ . If this information is available to the individual agents, the question is how he can get them to reveal it. This is the standard problem of mechanism design. If the information is not exogenously given to the individual agents, but must be acquired by them at a cost, there is the added issue of their incentives for information acquisition. The present paper posits that - initially at least - neither the designer nor the agents know the values of  $\omega_i^j$ . The agents may learn some of their own valuations  $\omega_i^j$ . Therefore a mechanism sets incentives not only for the allocation of houses for given (ex post) preferences, but also for the acquisition of information on houses.

The values of  $\omega_i^j$  are assumed to be drawn from binary distributions that are identical across agents. The draws are independent across agents and across houses. Formally, for any agent any house  $h_i$  has high value  $A_i$  with probability  $p_i$  and low value  $a_i$  with the complementary probability. This assumption of independent draws is essential to rule out informational externalities.

Agents 1 through  $n - 1$  can learn their (idiosyncratic) value of exactly one house each (**investigate** one house). There is no explicit cost of learning in the model. An agent who opts to learn the value of some house  $h_i$  only faces the opportunity cost of not being able to investigate any other house.

The  $n$ th agent cannot investigate any house.<sup>3</sup>

Formally, an agent's type is identified with the vector of values he assigns to the houses  $\omega^j = (\omega_i^j)_{i=1,\dots,n}$ . The set of all possible types of agent  $j$  is denoted by  $\Omega^j$ . The underlying state space of the model is  $\Omega = \Omega^1 \times \dots \times \Omega^n$ , the space of all possible profiles of values  $\omega = (\omega^1, \dots, \omega^n) = (\omega_i^j)_{i,j=1,\dots,n}$ . The distribution of states is denoted by  $P$ , and can be constructed from the assumptions on the distributions of the separate values  $\omega_i^j$ . The space of a posteriori preference profiles  $\bar{\Omega}$  is defined as the space of all  $n \times n$  matrices  $\bar{\omega}$  with the feature that  $\bar{\omega}_i^j = A_i$ , or  $\bar{\omega}_i^j = a_i$  for at most one  $i$  for each  $j < n$ , and  $\bar{\omega}_i^j = \alpha_i$  for all other  $i, j$ . If house  $h_i$  was investigated by agent  $j$  then  $\bar{\omega}_i^j \neq \alpha_i$ .<sup>4</sup>

### 3.2 Allocation Mechanisms

A **mechanism**  $\Gamma = (R, g)$  is a collection of  $n$  strategy sets  $R := (R^1, \dots, R^n)$  and an outcome function  $g : R \rightarrow M$  that maps every strategy profile  $r := (r^1, \dots, r^n)$  to an allocation in  $M$ . While this definition of a mechanism is standard, the function that such a mechanism serves in the context of the present setup differs from the standard case. In the standard case, agents know their types  $\omega^j$  and condition their strategies on these. In the present case, types arise endogenously out of learning choices. Said differently: the same mechanism  $\Gamma = (R, g)$  induces different games in cases where agents know their types and the alternative cases, in which types can be learned at a cost. In the latter case, the extensive form game induced by  $\Gamma = (R, g)$  needs to be augmented by nodes that reflect the learning decisions of agents.

In the case of endogenous information acquisition, a mechanism  $\Gamma = (R, g)$  induces an extensive form game  $(T, j, D, \mathcal{I})$  where  $T$  denotes the set of

---

<sup>3</sup>The significantly more complicated case, in which all  $n$  agents can investigate a house is discussed in the Appendix (Section 10.2).

<sup>4</sup>This notion of a posteriori preferences corresponds to the moment right after the investigations. Of course, once the agents occupy houses, they will find out whether they like their respective houses. If one was to study re-trading of houses, one would probably want to detail more stages information acquisition, assuming always that an occupant of a house would learn the value of the house he is living in.

nodes,  $j$  and  $D$  are functions such that  $j(t)$  is the agent who gets to choose from a action sets  $D(t)$  at node  $t$  and  $\mathcal{I}$  describes the information sets as follows:

The game tree starts off with a move of nature in which the state  $\omega$  is drawn from  $P$ . The remaining nodes  $T$  can be partitioned into set of learning nodes  $T_l$ , rule nodes  $T_R$ , and terminal nodes  $T_\tau$ . At any learning node  $t \in T_l$ , agent  $j(t)$  gets to choose whether and which house to investigate, so  $D(t) = \{\emptyset, l : h_1, l : h_2, \dots, l : h_n\}$  for  $t \in T_l$ , where the  $l : h_i$  stands for the choice to investigate house  $h_i$ , and  $\emptyset$  stands for the choice not to investigate a house. The set of rule nodes  $T_R$  is determined by the mechanism  $\Gamma$ . The game is structured such that rule node  $t$  is prefaced with a learning node for agent  $j(t)$ , if  $j(t)$  has not yet investigated a house on the path leading up to  $t$  and if  $j(t) < n$ . It is useful to go back to the sketch of the game tree induced by serial dictatorship in Figure 2. Agent 1's first node after the draw of nature is a learning node. Any learning node is succeeded by three rule nodes  $t^\alpha, t^\beta$ , and  $t^\gamma$  for agent 1 (note that due to restrictions of space I only drew the subtree following agent 1's choice to investigate  $h_1$ ). In each of these three rule nodes, agent 1 faces the same decision ( $D(t) = \{h_1, h_2, h_3\}$  for  $t = t^\alpha, t^\beta$  and  $t^\gamma$ ) as prescribed by the mechanism of serial dictatorship.

Some information sets  $\mathcal{I}$  are directly determined by  $\Gamma = (R, g)$ : if the rules set by the mechanism designer are such that agent 3 knows which action agent 2 chose at some preceding rule node, the information sets need to reflect this knowledge. In addition, information sets  $\mathcal{I}$  are such that learning is private, in the sense that only an agent knows whether or which house he investigated, and he is also the only one to know the outcome of that investigation.<sup>5</sup>

The terminal nodes specify the payoffs of agents, which are calculated as follows. The decisions in the rule nodes of the path leading up to some terminal node  $t \in T_\tau$  determine a matching  $\mu$  of houses to agents. The payoff to agent  $j$  is the a posteriori value he assigns to the house he has been allocated. The a posteriori value that agent  $j$  assigns to house  $h_i$  at some

---

<sup>5</sup>This assumption implies that the placement of the learning nodes directly before the relevant rule nodes is without loss of generality. Given that agents can communicate only through the channels given by the mechanism, there is no strategic benefit to the anticipation of learning decisions.

terminal node  $t \in T_\tau$  is  $\alpha_i$ , if the agent did not investigate that house on the path leading up to  $t$ . If  $j$  did investigate his assigned house, his value is  $\omega_i^j$ , where the first branch of the tree leading up to  $t$  is  $\omega$ .

Consider the left-most terminal node in the sketch in Figure 2. The allocation in this terminal node is determined through agents 1's choice of  $h_1$  and agent 2's choice of  $h_2$  in their respective rule nodes. According to the state of nature  $\omega$ , each agent values each house highly. However, according to the learning decisions, only agent 1 knows the value of the house he is being assigned. So the vector of a posteriori values is  $(A_1, \alpha_2, \alpha_3)$ .

A strategy of agent  $j$  in the given extensive form game is denoted by  $s^j \in S^j$ , where  $S^j$  denotes the set of his strategy profiles. Strategy profiles and sets of strategy profiles are denoted by  $s = (s_1, \dots, s_n)$  and  $S := (S_1, \dots, S_n)$ , respectively.

### 3.3 Learning Trees and Allocation Functions

For any mechanism  $\Gamma = (R, g)$  and the corresponding game with endogenous information acquisition, I define two functions describing the learning and allocations for given states and strategy profiles:  $g^l : S \times \Omega \rightarrow \overline{\Omega}$  and  $g^F : S \times g^l(S \times \Omega) \rightarrow M$ . The function  $g^l : S \times \Omega \rightarrow \overline{\Omega}$  maps strategy profiles and states to a posteriori preference profiles. For a house  $h_i$  that has been investigated by some agent  $j$  under some strategy profile at state  $\omega$ , we have that  $\overline{\omega}_i^j = \omega_i^j$ ; for all other pairs of agents and houses, we have that  $\overline{\omega}_i^j = \alpha_i$ . The function  $g^F$  describes the allocation determined by the mechanism for given a posteriori preferences  $\overline{\omega} \in g^l(S \times \Omega) \subset \overline{\Omega}$  and given strategy profiles  $s$ .<sup>6</sup>

To analyze the learning behavior and allocations implemented by mechanisms, I next define learning trees and allocation functions.

**Definition 1** *For a given housing problem, a **learning tree** and **allocation function**  $(l, F)$  is defined as follows:*

*A **learning tree** is a function  $l : \Omega \rightarrow \overline{\Omega}$  whose values are determined us-*

---

<sup>6</sup>Since allocations can only depend on the known values, the allocation function  $g^F$  maps  $g^l(S \times \Omega) \subset \overline{\Omega}$  instead of  $\Omega$  to  $M$ .

ing a rooted binary tree with house-agent-pairs  $(h_i, j)$  as the nodes and values  $A_i$  and  $a_i$  as the edges with the following properties. The root node  $(h_{i_r}, j_r)$  prescribe agent  $j_r$  to start off the learning process with an investigation of  $h_{i_r}$ . From then on, learning follows (\*)

(\*) If the value is high and the node has an edge representing the high value, learning continues with the house-agent-pair prescribed by that node. and (\*) is repeated. If the node has no such edge, learning terminates. The same holds for low values.

The tree has the property that there is no path on which the same agent appears twice. Agent  $n$  never appears on the tree. The a posteriori profile of preferences  $l(\omega) = \bar{\omega}$  is such that  $\bar{\omega}_i^j = \omega_i^j$  holds for any node  $(h_i, j)$  that is visited for  $\omega$ ; otherwise we have  $\bar{\omega}_i^j = \alpha_i$ .

Any function  $F: l(\Omega) \rightarrow M$  is called an **allocation function**.

The procedural description reflects the fact that the choice to investigate some house can only depend on the outcomes of *prior* investigations. The assumptions that no agent can investigate two houses and that agent  $n$  cannot investigate any house are reflected in the condition on the paths of the tree. The condition that agents can only learn true values determines the calculation of the values of  $l$ . Note that the domain of the allocation function depends on the learning tree; learning trees and allocations functions are therefore defined as pairs. House  $h_i$  is assigned to agent  $j = F(l(\omega))(h_i)$  at state  $\omega$ , if the learning tree and allocation function are  $(l, F)$ .

From now on, I restrict attention to learning trees  $l$ , according to which agents move in the order of their index. This makes sense, since the identity of the agents who learn houses has no relevance for total welfare.<sup>7</sup> All uniqueness results in the sequel are understood modulo renaming of agents.

Any mechanism  $\Gamma$  together with a strategy profile  $s$  can be associated with a learning tree  $l$  and an allocation function  $F$ , through  $l(\omega) = g^l(s, \omega)$  and  $F(\bar{\omega}) = g^F(s, \bar{\omega})$ . For the particular case that the strategy profile under

---

<sup>7</sup>This is implied by the assumption that agents are a priori identical, in the sense that their preferences are independent draws from identical distribution, together with the assumption that each agent can investigate at most one house.

consideration is a perfect Bayesian equilibrium, we say that the associated  $(l, F)$  are implemented by  $\Gamma$ .

**Definition 2** *The mechanism  $\Gamma$  **implements**  $(l, F)$  (via  $s^e$ ) if the game induced by  $\Gamma$  has a perfect Bayesian equilibrium strategy profile  $s^e$ , such that  $l = g^l(s^e, \cdot)$  and  $F = g^F(s^e, \cdot)$  on  $g^l(\cdot, \Omega)$ .*

Note that  $\Gamma$  might have multiple equilibria. Therefore, one can use the same mechanism  $\Gamma$  to implement different learning trees and allocation functions. At times, it is useful to single out the equilibrium strategy profile  $s^e$  for which  $l = g^l(s^e, \cdot)$  and  $F = g^F(s^e, \cdot)$  holds, in which case  $\Gamma$  is said to implement  $(l, F)$  via  $s^e$ . Also note that a concept of equilibrium in *pure* strategies is used.

The designer's goal to choose a mechanism that maximizes welfare can now be expressed as:

$$\begin{aligned} & \max W(l, F) \\ & \text{s.t. } \exists \Gamma : \Gamma \text{ implements } (l, F). \end{aligned}$$

where

$$W(l, F) := \sum_{\omega \in \Omega} P(\omega) \left( \sum_{i=1}^n \omega_i^{F(l(\omega))(h_i)} \right)$$

is defined as total expected welfare for the given learning tree  $l$  and allocation function  $F$ . The **first best learning tree and allocation function**  $(l^*, F^*)$  is defined as the learning tree and allocation function that maximize social welfare  $W(l, F)$ , ignoring implementability. The **second best learning tree and allocation function**  $(l^\circ, F^\circ)$  is defined as the learning tree and allocation function that maximize social welfare, conditional on  $(l^\circ, F^\circ)$  being implementable.

## 4 The First-Best Learning Process and Allocation Function

The main statistic needed to characterize the first best learning tree and allocation function  $(l^*, F^*)$  is the expected increase in welfare associated with

the investigation of a house when the designer is free to allocate that house at its expected value or at its observed value. Formally, the **option value**  $O_i$  of house  $h_i$  is defined as  $O_i = p_i(A_i - \alpha_i)$ . I assume from now on that there are no two houses with the same option value:  $O_i \neq O_{i'}$  for all  $i \neq i'$ .<sup>8</sup> I assume, in addition, that houses are indexed in terms of their option value such that  $O_i > O_{i+1}$  for all  $i$ .

**Lemma 1** *Any first-best learning tree and allocation function  $(l^*, F^*)$  can be described as follows:*

- *after any history of investigations,  $l^*$  prescribes to investigate a house with maximal option value among all houses which have not yet been found to be of high value in some prior investigation. This process continues until agent  $n - 1$ .*
- *$F^* : l^*(\Omega) \rightarrow M$  matches  $h_i$  with agent  $j$ , if agent  $j$  found house  $h_i$  to be of high value. All other houses are assigned to agents who evaluate them at their expected values.*

According to  $l^*$ , agents should investigate  $h_1$  as long as no one has found it of high value. Once this happens they should move on to the  $h_2$  and so forth. Observe that  $F^*$  is well-defined, since for any  $\bar{\omega} \in l^*(\Omega)$  at most one agent knows a particular house to be of high value. If we drop the assumption that no two houses have the same option value, there are as many optimal learning trees as there are ways to order the houses according to their option value.<sup>9</sup> The set of efficient allocation functions is large, since there are many different ways to assign houses to agents who did not investigate them.

Here I just discuss some of the main features of the proof, which can be found in the Appendix. In the proof, I define a set of **learning processes** and **allocation functions**  $(\tilde{l}, \tilde{F})$  that prescribe continuation strategies for

---

<sup>8</sup>This assumption is not without loss of generality. However, while the results only change slightly, the notation becomes significantly less cumbersome. Throughout the text, I will point out how the results need to be amended to account for the case that  $O_i \neq O_{i'}$  holds for some  $i \neq i'$ .

<sup>9</sup>This presumes, of course, that agents move in order of their index, as discussed at the end of Section 3.

all histories of investigations. The upshot is that backward induction can be used to characterize the welfare-optimal learning process and allocation function  $(\tilde{l}^*, \tilde{F}^*)$ .

I start by showing that  $\tilde{F}^*(\bar{\omega}) = \sum_{i \in N} \bar{\omega}_i^{\mu(i)}$  with  $\mu(i) = \tilde{F}^*(\bar{\omega})(h_i)$  maximizes welfare if each house  $h_i$  is matched to an agent  $\mu(i)$  who assigns value  $\max_j \{\bar{\omega}_i^j\}$  to this house - conditioning on  $\bar{\omega}$  being the profile of a posteriori preferences.

To prove the optimality of  $\tilde{l}^*$  by backwards induction, consider the  $n - 1$ st learner. Observe that the assumption that the  $n$ 'th agent cannot investigate any house and therefore evaluates all houses at their expected value implies that no matter which house the  $n - 1$ st agent investigates the designer is free to allocate each house either at its expected value or to some agent who investigated it.<sup>10</sup> Therefore, the  $n - 1$ st agent should learn the house with the highest option value that has not yet been found to be of high value by any agent. The inductive step follows from similar arguments.

## 5 Sequential Revelation Mechanisms

The original version of the revelation principle as formulated by Myerson (1982, 1985) does not apply to games of endogenous sequential learning.<sup>11</sup> I therefore start this section with the statement of a version of the revelation principle that applies to the present case of sequential learning. To this end, I define a mechanism as a **sequential revelation mechanism** if each agent has at most one rule node on any path of the corresponding game tree, if each rule node has two edges and if these edges correspond to the messages “ $A_i$ ” and “ $a_i$ ”. Formally, we have that  $D(t) = \{A_{i(t)}, a_{i(t)}\}$  for some house  $i(t)$  for any  $t \in T_R$ , where the set  $D(t)$  is to be interpreted as the set of

---

<sup>10</sup>The assumption that only  $n - 1$  agents can investigate a house is essential for this argument. This problem renders the extension to the case of  $n$  learners non-trivial. Since the terminology developed in the proof Lemma 1 turns out to be useful in the discussion of the  $n$ -learner case, this discussion is relegated to the Appendix (Section 10.2).

<sup>11</sup>To see this, consider the welfare-optimal mechanism as defined in the introductory example (Section 2.5). There is no normal form game in which agents 1 and 2 simultaneously announce their types to the designer, while at the same time agent 2 conditions his learning about his own type on his knowledge of the type of agent 1.

possible messages that can be sent to the mechanism designer. A strategy profile in the corresponding game with endogenous information acquisition  $s^t$  is considered **truthful** if the following two conditions hold: firstly, in any learning node  $t \in T_l$ , agent  $j(t)$  investigates house  $h_i$ , if the choices in the succeeding rule nodes  $t'$  are  $D(t') = \{A_i, a_i\}$ . Secondly, an agent who has investigated the house prescribed by  $s^t$  announces  $A_i$  if and only if he found  $h_i$  to be of high value.

Observe that in the context of endogenous learning truthfulness not only requires that agents report their types. Agents are, in addition, required to obtain the information that they are supposed to report truthfully. If an agent investigates some house  $h_i$  before a rule node  $t$  with  $D(t) = \{A_i, a_i\}$ , he cannot possibly report his type truthfully. In that case, he evaluates  $h_i$  at  $\alpha_i \notin \{A_i, a_i\}$ .<sup>12</sup>

A sequential revelation mechanism  $\Gamma = (R, g)$  **truthfully** implements a learning tree and allocation function  $(l, F)$  if the truthful strategy profile  $s^t$  is an equilibrium in the induced game and if  $g^l(\cdot, s^t) = l$  and  $g^F(\cdot, s^t) = F$ .

**Example 1** Reconsidering the housing problem defined in Section 2, observe that serial dictatorship is not an sequential revelation mechanism, as the first dictator can choose from three houses. However, the learning tree and allocation function  $(l, F)$  that are implemented by serial dictatorship can be implemented by the following sequential revelation mechanism: first, the mechanism designer asks the first agent whether he values  $h_1$  highly or not. If the first agent has claimed to value  $h_1$  highly, the designer asks the second agent for his valuation of  $h_2$ . Allocations are such that agent 1 is assigned  $h_1$  if he claims to value it highly. In that case, house  $h_2$  is assigned to agent 2, if and only if he claims to value it highly; otherwise he is assigned  $h_3$ . If agent 1 claims  $a_1$ , agents 1,2 and 3 are allocated  $h_2, h_1$ , and  $h_3$  in that order. Another example of a sequential revelation mechanism was already discussed

---

<sup>12</sup>Of course, sequential revelation mechanisms could also have been defined using larger sets of announcements  $D(t) = \{A_1, a_1, A_2, a_2, \dots, A_n, a_n\}$ . In such mechanisms agents can truthfully report their type after any learning decision. The upshot of “tailoring” sequential revelation mechanism for particular learning trees is that smaller strategy sets facilitate implementation: with larger strategy sets, more deviations need to be considered to establish that a particular strategy profile is an equilibrium.

in Section 2.5.

The example is generalized in the next theorem on sequential revelation mechanisms.

**Theorem 1** *A learning tree and allocation function  $(l, F)$  are implementable by some mechanism  $\Gamma$ , if and only if they are truthfully implementable by some sequential revelation mechanism  $\Gamma'$ .*

**Proof** Let  $(l, F)$  be implemented by  $\Gamma$  via  $s^e$ . Prune the game tree, so that all edges that correspond to actions never chosen according to  $s^e$  are deleted. Next, drop all remaining rule nodes with only one edge and observe that the surviving rule nodes have the feature that the action chosen in these nodes depends on the investigation of a house. Consequently, there is at most one rule node remaining per path and player. All these rule nodes have exactly two branches, since the state variable on which the choice is based is binary and since only pure strategies are considered. The suitably pruned strategy profile  $s_p^e$  is also an equilibrium in the pruned tree and corresponding mechanism  $\Gamma'$  and implements  $(l, F)$ . Now construct a mechanism  $\Gamma''$  that has the same set of nodes  $T$  as  $\Gamma'$ . Construct  $\Gamma''$  through a relabeling of the actions in  $\Gamma'$  such that for any rule node  $t \in T_R$  at which  $s_p^e$  prescribes that  $j(t)$  learn  $h_i$  before  $t$  let  $D''(t) = \{A_i, a_i\}$  with the interpretation that  $D''(t)$  is the set of possible messages to the designer at node  $t$  in mechanism  $\Gamma''$ . The designer reveals information in  $\Gamma''$  according to the information sets in  $\Gamma'$  and determines outcomes in  $\Gamma''$  as if the agents had played according to  $s_p^e$  in the game induced by  $\Gamma'$ . Observe that  $(l, F)$  is implemented in  $\Gamma''$  via the strategy profile that corresponds to  $s_p^e$ . The reverse conclusion is trivial, since sequential revelation mechanisms are mechanisms, too.  $\square$

Theorem 1 can easily be extended to distributions with more than two values: the number of available messages simply needs to be adjusted to the number of relevant states. Furthermore, the assumption that values of all houses are distributed independently and identically across agents did not enter the proof. Similarly, the theorem can be amended to multiple learning

decisions. The main difference with respect to the standard revelation principle is that in the present case, it is not possible to elicit the agents' types simultaneously. In the present case, types are endogenous and some agents might condition their learning on the actions of others. Consequently, some agents might have to wait for the revelation of some other agents' types to base their endogenous learning decision on this information. Furthermore, the designer not only has to provide proper incentives to reveal the truth but he also has to provide the incentive to learn according to his desired learning tree.

## 6 Implementation of the First-Best

Lemma 1 and Theorem 1 lead up to Corollary 1 on the set of cases in which the first best is implementable. To understand the problems associated with the implementation of the first-best learning tree and allocation function, consider a sequential revelation mechanism  $\Gamma$  that implements  $(l^*, F^*)$ . The learning tree  $l^*$  prescribes the order in which agents are supposed to learn houses. Since  $\Gamma$  is a sequential revelation mechanism, the choice offered to agent  $j$  at a node where he is supposed to investigate house  $h_i$  must be a choice between declaring  $A_i$  or  $a_i$ . Next,  $F^*$  prescribes that the agent obtains  $h_i$  if he (truthfully) declares  $A_i$ . The fall-back option, that is, the lottery over houses that declaring  $a_i$  corresponds to, must satisfy the following (potentially conflicting) three conditions. The probability that the agent obtains  $h_i$  if he announces  $a_i$ , must first of all, be zero; otherwise  $F^*$  would be violated. Secondly, for truth-telling to be equilibrium behavior, agent  $j$  must find it in his best interest to investigate  $h_i$  and to report the true value when given the choice between obtaining  $h_i$  and the fall-back option. Thirdly, the fall-back options must be consistent in the sense that the mechanism terminates with allocations.

From this discussion it can already be gleaned that the designer's ability to design fall-back options is of essence for the implementation of the first-best. To grasp the requirement better that an agent should find it in his best interest to investigate  $h_i$  when  $l^*$  prescribes for him to investigate  $h_i$ , some more definitions are needed. Lotteries on houses are defined as  $\pi :=$

$(\pi_1, \dots, \pi_n)$ , where  $\pi_i$  stands for the probability that the agent obtains  $h_i$ . Degenerate lotteries  $\pi$  with  $\pi_i = 1$  for some  $i$  are denoted by  $h_i$ . The expected value of the lottery  $\pi$  is denoted by  $E(\pi)$ .

**Definition 3** A lottery  $\pi$  is said to *incentivize* (the learning of)  $h_i$  if given the choice between  $h_i$  and  $\pi$ , the investigation of  $h_i$  is strictly preferable to no investigation and weakly preferable to the investigation of any other house. If  $\pi$  incentivizes  $h_i$ , I write  $h_i \succsim_l \pi$ .<sup>13</sup>

Observe that statements such as  $\pi' \succsim_l h_i$  or  $\pi' \succ \pi$  are left undefined for non-degenerate lotteries  $\pi'$ . For any  $h_1 = (p_1, A_1, a_1)$  and  $h_2 = (p_2, A_2, a_2)$  the requirement of  $h_1 \succsim_l h_2$  amounts to the following inequalities:

$$\begin{aligned} A_1 &> \alpha_2 > a_1 \\ p_1 A_1 + (1 - p_1) \alpha_2 &\geq p_2 A_2 + (1 - p_2) \alpha_1 \end{aligned}$$

The first inequality ensures that learning  $h_1$  is strictly preferred to choosing without learning. The second inequality formalizes the requirement that learning  $h_1$  is weakly preferred to learning  $h_2$ . If an agent is given the option to choose a house from a set  $\{h_1, h_2\}$  with  $h_1 \succsim_l h_2$  it is optimal for him to investigate  $h_1$  and to choose it, if and only if it has high value for him.

The relation between the learning order  $\succsim_l$  and the order by option values describes the crux of the present design problem. If the two coincide the problem of the designer is straightforward, and agents are happy to investigate houses in the order prescribed by the first-best learning tree. The problem lies in the fact that these two relations do not generally coincide. It is convenient to introduce further notation for sequential revelation mechanisms. Let  $d_A(t)$  be the number of high value statements in the history leading up to  $t$ . Define a function  $i : T_R \rightarrow \{1, \dots, n\}$  such that at node  $t$  the designer asks agent  $j(t)$  for the value of house  $h_{i(t)}$ . For any strategy profile  $s$ , any rule-node  $t \in T_R$  and the two possible announcements  $X \in \{A_{i(t)}, a_{i(t)}\}$  at  $t$ , let  $\pi_i(t, s, X)$  be defined as the probability with which agent  $j(t)$  obtains  $h_i$ , given that he

---

<sup>13</sup>This is equivalent to demanding that the investigation of  $h_i$  be strictly preferable to no investigation and weakly preferable to the investigation of any house *in the support of*  $\pi$ , as investigating a house not in the support of  $\pi$  corresponds to not investigating any house.

makes announcement  $X$  at node  $t$ , and given that all other agents follow the strategy profile  $s$ . Let  $\pi_i(t, s, X) = (\pi_1(t, s, X), \dots, \pi_n(t, s, X))$  stand for the corresponding lottery over houses.

**Corollary 1** *The first-best  $(l^*, F^*)$  is implementable, if and only if there exists a sequential revelation mechanism  $\Gamma$ , such that the following holds for all rule-nodes  $t \in T_R$ :*

- $i(t) = d_A(t) + 1$ .
- $\pi(t, s^t, A_{i(t)}) = h_{i(t)}$  and  $\pi_i(t, s^t, a_{i(t)}) = 0$ .
- $h_{i(t)} \succeq_l \pi(t, s^t, a_{i(t)})$ .<sup>14</sup>

**Proof** By Theorem 1,  $(l^*, F^*)$  can be implemented, if and only if it can be implemented truthfully through a sequential revelation mechanism. Let  $\Gamma = (R, g)$  be the sequential revelation mechanism that truthfully implements  $(l^*, F^*)$ . Observe that  $l^* = g^l(\cdot, s^t)$  holds, if and only if the first requirement above holds. Next, observe that conditional on the first observation,  $F^* = g^F(\cdot, s^t)$  holds, if and only if the second requirement holds. Finally  $s^t$  is a perfect Bayesian equilibrium, if and only if the last requirement holds, since  $\pi(t, s^t, A_{i(t)}) = h_{i(t)}$  holds for all nodes  $t$ .<sup>15</sup>  $\square$

Observe that Corollary 1 can be split into a “mechanical” and an “artful” part. The first two conditions straightforwardly determine some properties of any sequential revelation mechanism that might implement  $(l^*, F^*)$ : the designer has to offer the houses that should be learned according to  $l^*$  in the sequence prescribed by  $l^*$ . The art of mechanism design stands out when considering the third requirement, which demands that there is a way to use the houses that are not found to be of high value as fall-back options to incentivize learning according to  $l^*$ . The following example defines a housing problem in which the first best cannot be implemented.

---

<sup>14</sup>If some houses have the same option value, the statement has to hold for some indexation of the set of houses with  $O_i \geq O_{i+1}$  for all  $i$ .

<sup>15</sup>Note that all nodes are reached with positive probability; therefore, we need not be concerned with out of equilibrium beliefs. If I had defined sequential revelation mechanisms with message spaces large enough to allow for the announcement of any possible a posteriori type, as suggested in footnote 12, this would not hold.

**Example 2** Define a housing problem with three agents through  $h_1 = (p_1, A_1, a_1) = (\frac{1}{2}, 70, -10)$ ,  $h_2 = (p_2, A_2, a_2) = (\frac{3}{4}, 100, 0)$ , and  $h_3 = \alpha_3 = -2$ . Observe that  $\alpha_1 = 30$ ,  $\alpha_2 = 75$ ,  $O_1 = 20$ , and  $O_2 = \frac{75}{4}$ , which implies that houses are indexed by their option values. To see that the first-best is not implementable, suppose it was. Let  $\Gamma = (R, g)$  be the sequential revelation mechanism that implements it according to Theorem 1. Since  $h_1$  has the highest option value, the first agent needs to be given the choice to announce either  $A_1$  or  $a_1$ . Let us have a look at the node after a declaration of  $A_1$  by the first agent. We must have that  $\pi(A_1, s^t, A_2) = h_2$ , since agent 2 has to be allowed to keep  $h_2$ , if he would like to do so. On the other hand, we must have  $\pi(A_1, s^t, a_2) = h_3$ , as  $h_3$  is the only other remaining house. However, agent 2 is not willing to investigate  $h_2$  in this node, as  $h_2 \succ_l h_3$  does not hold (since  $a_2 > \alpha_3$ ).

What is important to note is that the inefficiency of the example is not one of asymmetric information. To see this fix any knowledge structure consistent with the example, meaning assume that agents 1 and 2 each privately know the value of one house. For each of these 3 cases (either both know the value of  $h_1$ , or both know the value for  $h_2$ , or each knows the value of a different house) one can design mechanisms that implements a welfare maximal allocation. With endogenous learning, though, it is not possible to design a mechanism that achieves the first best ex ante welfare. The difference between the ex ante optimal welfare and the welfare due to random assignment is  $p_1 O_1 + p_1(1 - p_1)O_1 + (1 - p_1)O_2$ , I show in Section 8 that the difference between the highest implementable ex ante welfare and the welfare associated with random assignments is just  $p_1 O_1 + p_2 O_2$ .

## 7 Trading Mechanisms

Can all welfare optima of housing problems with endogenous information acquisition be reached via markets? The second welfare theorem tells us that in an environment with divisible goods, convex and non-satiated preferences (and without endogenous information acquisition), any Pareto optimum can be sustained as a market equilibrium with transfers. Similarly, for the case of

housing problems (without endogenous information acquisition), any Pareto optimum can be reached via the choice of some appropriate initial allocation and free trade among agents (see, for instance, Abdulkadiroglu and Sonmez (1998)). Here I will show that endogenous information acquisition breaks this result. In fact, I show that not even the second best learning tree and allocation function needs  $(l^\circ, F^\circ)$  to be implementable through free trade.<sup>16</sup>

In this section, I identify “free trade” for the given environment with Gale’s top trading cycles mechanism as defined by Shapley and Scarf (1974). In this mechanism, each agent initially owns one house. In a first round, all agents are invited to point to the owners of their most preferred houses. Since there are only finitely many agents, at least one cycle forms. All agents in these cycles are assigned the houses that they are pointing to. The same procedure is repeated until all agents have been assigned a house. This mechanism can be considered a “free-trade” mechanism, since property rights for all houses are assigned to agents and since all exchanges are voluntary (agents may always point to their own house and form a cycle of length one).

**Theorem 2** *The second-best learning tree and allocation function  $(l^\circ, F^\circ)$  might not be implementable through Gale’s top trading cycles mechanism.*

**Proof** Reconsider the introductory example. I already showed in Section 2.5 that there exists a mechanism that does implement the first-best  $(l^*, F^*)$ . Next, I show that this is the unique sequential revelation mechanism that implements  $(l^*, F^*)$ . To see this, consider a sequential revelation mechanism that does implement  $(l^*, F^*)$  and observe that the designer has very little leeway in designing  $F^*$ . For the any a posteriori preferences profile  $\bar{\omega}$  with  $\bar{\omega}_1^1 = A_1$ , agent 1 must be assigned  $h_1$ . The welfare-optimal allocations are then uniquely determined by the outcome of agent 2’s investigation of  $h_2$ . If he finds  $h_2$  of high value, then  $h_2$  and agent 2 must be matched; if not,  $h_3$  must be allocated to agent 2. On the other hand, following agent 1’s observation that he values house  $h_1$  at  $a_1$ , agent 2 must investigate  $h_1$  to follow  $l^*$ . To have an incentive to do so and to be in line with a welfare-

---

<sup>16</sup>Example 2 entails that we do not even need to bother with the implementation of the first best through trade, as it was shown that no mechanisms whatsoever implements the first best in that example.

optimal allocation, agent 2 must be given a choice between  $h_1$  and  $h_2$ . This implies that the allocation must be such that agent 1 obtains  $h_3$  and agent 2 obtains  $h_2$  after both agent 1 and 2 found  $h_1$  to be of low value. Finally, to give agent 1 an incentive to investigate  $h_1$ , the fall-back option  $\pi$  he receives when announcing  $a_1$  must be such that  $h_1 \succ_l \pi$ . This implies that the allocation for the a posteriori preference profile, according to which agent 1 found  $h_1$  to be of low value, while agent 2 found the same house of high value, must be such that agent 1 obtains  $h_2$  and agent 2 obtains  $h_1$ . This completes the proof that the mechanism described in Section 2.5 is the unique sequential revelation mechanism that implements  $(l^*, F^*)$ .

Now suppose that there was an initial assignment of houses such that Gale's top trading cycles mechanism implemented  $(l^*, F^*)$ . By the above arguments, the equivalent sequential revelation mechanism must be the one described in Section 2.5. But we know that each agent has a positive chance to be assigned  $h_3$  according this mechanism. But this stands in conflict with the observation that there are no a posteriori preferences, under which the initial owner of  $h_1$  would be willing to trade that house for  $h_3$ . We can conclude that in the introductory housing problem the second-best  $(l^\circ, F^\circ)$  (which is equal to the first-best  $(l^*, F^*)$ ) cannot be implemented through Gale's top trading cycles mechanism.  $\square$

One might criticize Gale's top trading cycles mechanism as too limited a notion of "free trade". Gale's mechanism does, for example, rule out that some agents start out by owning multiple houses. The class of hierarchical exchange mechanism as defined by Papai (2000) is not vulnerable to this criticism. It turns out that Theorem 2 also holds true for this much larger class of mechanisms. Since the definition of hierarchical exchange mechanism is cumbersome I restrict myself to a rather informal discussion of this result here. A mechanism is a **hierarchical exchange mechanism** if all houses are initially owned by some agents and if the final allocation is determined through inheritance rules and voluntary trading. Initially some agents might own more than one house; others might not own any houses. Most importantly, though, each house starts out being owned by someone. The mechanism requires that each agent points to some house and houses point to their owners. Due to the finiteness of the problem, there is at least one

group of agents and equally many houses that form a cycle. All agents and houses in that cycle are assigned the house that they point to. If an owner of several houses takes part in such an exchange ring, one of his houses will be assigned through the cycle. All other houses are passed on to agents who have not yet been assigned a house according to some fixed inheritance rule. Then the procedure restarts with the remaining agents pointing to the remaining houses. The finiteness of the problem implies that all houses will be assigned through this procedure.

Serial dictatorship and Gale's top trading cycles are both examples of hierarchical exchange mechanisms. According to serial dictatorship, all houses are initially owned by the first dictator. The inheritance rule prescribes that all houses that he does not choose (by forming a cycle of length one, by pointing to one of his houses) are inherited by the second dictator and so forth. In Gale's top trading cycles, each agent initially owns one house, inheritance rules are consequently irrelevant.

Now to see that Theorem 2 cannot be fixed using the larger class of hierarchical exchange mechanisms, meaning that also the much stronger statement that the second-best learning tree and allocation function  $(l^\circ, F^\circ)$  might not be implementable through any hierarchical exchange mechanism, reconsider the proof of Theorem 2. Observe that this proof revolved around the observation that the following three ingredients are incompatible: 1. according to the unique first best learning tree and allocation function, any agent has a positive chance to obtain  $h_3$  as his assignment. 2. Some agent owns  $h_1$  in the initial period of the mechanism. 3. There are no ex-post preferences according to which the owner of  $h_1$  would voluntarily accept being assigned  $h_3$ . The same incompatibility applies to *any* hierarchical exchange mechanism.

This extension of Theorem 2 makes it hard to think of a version of the second welfare theorem that would hold for housing problems with endogenous information acquisition. In the case of known types, all Pareto optima can be reached through market mechanisms given suitable endowments: in this case, control of the designer can at most be a hindrance to achieving Pareto optima. Conversely, in the case of endogenous information acquisition, there are reasons to limit trading: such limits might be instrumental in the provision of the proper incentives for learning.

## 8 Trade-Offs: Simultaneous versus Sequential Learning; Efficient Learning versus Efficient Allocations

In this section, I show that the second-best learning tree and allocation function  $(l^\circ, F^\circ)$  might only be implementable through a mechanism in which agents acquire information simultaneously. Up to now, one might get the impression that sequential learning is always a boon for the designer: as long as all option values of the houses are different, the first-best learning tree prescribes that the choice of the house to be investigated next should *always* be conditioned on the outcomes of prior investigations. But this is just a feature of the *first-best* learning tree. I get back to Example 2 to show that for some housing problems it is strictly preferable to use a mechanism in which agents cannot condition their choice to learn a house on the outcomes of other investigations. Such simultaneity might be optimal, since it provides the mechanism designer with a source of randomness, which is, in turn, an important instrument to get agents to learn.

The same example can be used to illustrate the trade-off between efficient learning and efficient allocations, if the first-best  $(l^*, F^*)$  is not implementable. The search for the second-best  $(l^\circ, F^\circ)$  can be structured as, first, the calculation of a frontier in which more efficient learning trees are implemented together with less efficient allocation functions. Then welfare for all  $(l, F)$  on the frontier can be calculated to determine the second-best  $(l^\circ, F^\circ)$ .

**Example 3** Reconsider Example 2. Define a sequential revelation mechanism  $\Gamma' = (R', g')$  through the rule-tree given in Figure 5. I show that the mechanism  $\Gamma'$  implements the first-best learning tree  $l^*$  and achieves the highest welfare among all mechanisms that do so, formally,  $\Gamma'$  implements  $(l^*, F')$  such that  $W(l^*, F') \geq W(l^*, F'')$  for all implementable  $(l^*, F'')$ .

Consider the truthful strategy profile  $s^t$  and observe that  $g^l(\cdot, s^t) = l^*$ . To see that  $s^t$  is an equilibrium in  $\Gamma'$ , observe that, after the history  $A_1$ , agent 2 can choose between  $h_2$  and  $h_1$ . Since  $h_2 \succ_l h_1$ , it is in agent 2's best interest to learn the value of  $h_2$  and to reveal it truthfully. On the other hand, after history  $a_1$  agent 2 faces a choice between  $h_1$  and  $h_3$ . Since

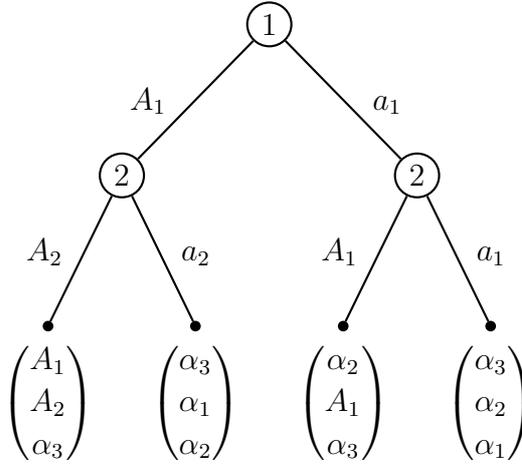


Figure 5: Rule-Tree for  $\Gamma'$

$h_1 \succsim_l h_3$ , the truthful strategy profile prescribes a best reply at this node. To analyze agent 1's decision, observe that the announcements of  $A_1$  and  $a_1$  correspond to the following lotteries over houses:  $\pi(\emptyset, s^t, A_1) = (\frac{3}{4}, \frac{1}{4}, 0)$  and  $\pi(\emptyset, s^t, a_1) = (0, \frac{1}{2}, \frac{1}{2})$ . The difference between the utility of investigating  $h_1$  and  $h_2$  can be expressed as follows:

$$\begin{aligned} & \frac{1}{2} \left( \frac{3}{4}70 + \frac{1}{4}(-2) \right) + \frac{1}{2} \left( \frac{1}{2}75 + \frac{1}{2}(-2) \right) - \\ & - \frac{3}{4} \left( \frac{1}{2}100 + \frac{1}{2}(-2) \right) + \frac{1}{4} \left( \frac{3}{4}30 + \frac{1}{4}(-2) \right) > 0. \end{aligned}$$

This implies, in turn, that learning  $h_1$  and truthfully revealing its value is a best reply for agent 1.

It only remains to be shown that there exists no other mechanism  $\Gamma''$  that implements some  $(l^*, F'')$  such that  $W(l^*, F'') > W(l^*, F')$ . To see this, observe that to get agent 2 to investigate  $h_2$  he has to be offered a choice between  $h_1$  and  $h_2$  (his choice has to be over deterministic outcomes, since he is the last one to investigate any houses). This implies that agent 1 can receive  $h_1$  at most with a probability of  $\frac{3}{4}$  if he finds it of high value. Next, observe that the necessary inefficiency just described is the *only* inefficiency in  $F'$ . Therefore  $W(l^*, F') \geq W(l^*, F'')$  holds for any implementable  $(l^*, F'')$ .

Now, of course, the second-best  $(l^\circ, F^\circ)$  need not have the feature that  $l^\circ = l^*$ . The next mechanism  $\bar{\Gamma} = (\bar{R}, \bar{g})$  implements a less efficient learning tree together with an efficient allocation function. Let  $\bar{R}^j = \{h_j, \text{no thanks}\}$  for  $j = 1, 2$  and let allocations be determined by the following matrix. The

	$h_2$	no thanks
$h_1$	$(h_1, h_2, h_3)$	$(h_1, h_3, h_2)$
no thanks	$(h_3, h_2, h_1)$	$(h_3, h_1, h_2)$

action “ $h_j$ ” can be interpreted as a choice of  $h_j$ , whereas the strategy “no thanks” can be interpreted as declining  $h_j$ . The mechanism has an equilibrium  $s^e$  which is such that agent  $j$  investigates  $h_j$  and keeps it, if and only if he found it of high value. To see that  $s^e$  is an equilibrium, observe that agent 1’s choice amounts to a choice among  $h_1$  and  $h_3$ . For agent 1’s fixed strategy  $s_1^e$ , agent 2 chooses from  $\{h_2, (\frac{1}{2}, 0, \frac{1}{2})\}$ , where  $h_2 \succ_l (\frac{1}{2}, 0, \frac{1}{2})$  holds as  $\alpha_2 > \frac{1}{2}70 + \frac{1}{2}(-2)$  and  $A_2 > E(\frac{1}{2}, 0, \frac{1}{2}) > a_2$ .<sup>17</sup>

The learning tree  $\bar{l} = \bar{g}^l(\cdot, s^e)$  is such that agent  $j$  unconditionally investigates house  $h_j$  for  $j = 1, 2$ , and therefore  $\bar{\omega} \in l(\Omega)$ , if and only if  $\bar{\omega}_i \neq \alpha_i$  for  $i = 1, 2$ . The allocation function  $\bar{g}^F(\cdot, s^e) = \bar{F}$  implemented by  $\bar{\Gamma}$  is such that any house that was found to be of high value by some agent is assigned at high value and all other houses are assigned at their expected values. Therefore,  $\bar{F}$  is an efficient allocation function.

To find the second-best  $(l^\circ, F^\circ)$  we do not need to look any further: we already found that  $F'$  is the allocation function that maximizes  $W(l^*, F)$  subject to  $(l^*, F)$  being implementable. Next, observe that  $\bar{l}$  differs from  $l^*$  only insofar as that  $l^*$  prescribes that  $h_1$  is investigated after  $h_1$  was found of low value, whereas  $\bar{l}$  prescribes that  $h_2$  is investigated in that same case. This is smallest possible deviation from  $l^*$ . Given that this learning tree  $\bar{l}$  is implementable together with an efficient allocation function  $\bar{F}$ , there cannot possibly be another implementable pair  $(\hat{l}, \hat{F})$  such that  $W(\hat{l}, \hat{F}) > \max(W(l^*, F'), W(\bar{l}, \bar{F}))$ .

---

<sup>17</sup>Observe that this is the only strategy profile that survives the iterated elimination of dominated strategies.

So we are facing a trade-off between efficient allocations and efficient learning here. We can either implement the efficient learning tree together with a less than efficient allocation function, or we can implement a less than efficient learning tree together with an efficient allocation function. In this particular case, the latter is preferable: the difference between  $W(l^*, F')$  and  $W(\bar{l}, \bar{F})$  can be expressed as the difference between  $p_1 p_2 (A_1 - \alpha_1) + p_1 p_2 (A_2 - \alpha_2) + (1 - p_1) p_1 (A_1 - \alpha_1) + (\alpha_1 + \alpha_2 + \alpha_3)$  and  $p_1 (A_1 - \alpha_1) + p_2 (A_2 - \alpha_2) + (\alpha_1 + \alpha_2 + \alpha_3)$  which comes down to  $O_1(p_2 - p_1) - O_2(1 - p_1) = 20(\frac{3}{4} - \frac{1}{2}) - \frac{75}{4}(1 - \frac{1}{2}) < 0$ .

An important observation to carry away is that  $h_1$  and  $h_2$  are being investigated simultaneously according to the second-best  $(l^\circ, F^\circ)$ . So while the first-best always requires that agents learn sequentially, such sequentiality might not be desirable to implement the second-best. In the present case,  $h_3$  is unattractive in any which way: its expected value is so low as to make it hard to use it to incentivize learning; moreover, its option value is zero. Therefore, to incentivize learning of both  $h_1$  or  $h_2$ , agents need to believe that the probability that they will end up with  $h_3$  is low. When learning is sequential, such belief cannot be generated.

In general there are two reasons why such randomness facilitates incentivization. First of all, the attractiveness to investigate a house increases with the probability that the agent would obtain this house if he found it if high value. Consider an environment in which each house attains its high value with probability  $\frac{1}{2}$  and in which house  $h_1$ 's high and low value are 100 and 0 whereas house  $h_2$  and  $h_3$ 's high and low value are 110 and  $-10$  each. Observe that it is optimal to learn the value of  $h_2$  for an agent who has the choice between  $h_1$  and  $h_2$ . Conversely, for an agent who can choose between  $h_1$  and an equal lottery on  $h_2$  and  $h_3$  it is optimal to learn the value of  $h_1$ . To understand the second reason, observe that a necessary condition for  $h_i \succsim h_{i'}$  to hold is that  $A_i > \alpha_{i'} > a_i$ . So if we have three houses  $h_1, h_2$  and  $h_3$  with  $\alpha_2 < a_1$  and  $\alpha_3 > A_1$  neither  $h_2$  nor  $h_3$  incentivize the learning of  $h_1$ . However,  $h_1 \succsim_l \pi$  can hold for a lottery  $\pi$  over  $h_2$  and  $h_3$  such that  $a_1 < E(\pi) < A_1$ . These two effects of randomization are summarized in the more general statement that the relation  $\succsim_l$  is convex, in the sense that

$h \succsim_l \pi$  together with  $h \succsim_l \pi'$  implies that  $h \succsim_l \rho\pi + (1 - \rho)\pi'$ .<sup>18</sup>

Also note that the calculation of the second-best  $(l^\circ, F^\circ)$  was broken down into the following two steps: I first calculated a frontier of learning trees and allocation functions that combined more efficient learning trees with less efficient allocation functions. I then compared the welfare achieved by the different pairs  $(l, F)$  on that frontier. Note that an application of this procedure to other problems requires a definition of the degree of efficiency of a learning tree. In the present case, this was not necessary as the learning tree  $\bar{l}$  is “obviously” the second-most efficient learning tree in the given example and as this second-most efficient learning tree is already implementable together with an efficient allocation function. In general, the notion of a more efficient learning tree could be operationalized as  $l$  is more efficient than  $l'$  if  $W(l, F) > W(l', F')$  for  $F$  and  $F'$  efficient allocation functions.

## 9 Conclusion

The present paper points out some major differences between mechanism design for housing problems with and without endogenous information acquisition. In the standard case, in which agents know their types, any desired Pareto optimum in a housing problem can be achieved through a trading

---

<sup>18</sup>To see this, consider a house  $h = (p, A, a)$  and two lotteries  $\pi, \pi'$  with  $h \succsim_l \pi$  and  $h \succsim_l \pi'$ . Since  $A > EU(\pi) > a$  as well as  $A > EU(\pi') > a$ , it follows that  $A > EU(\rho\pi + (1 - \rho)\pi') > a$  for all  $\rho \in (0, 1)$ . Moreover,

$$pA + (1 - p)EU(\pi) \geq p_{i_0}(\pi_{i_0}A_{i_0} + \sum_{i \neq i_0} \pi_i \alpha_i) + (1 - p_{i_0})\alpha \quad \text{and}$$

$$pA + (1 - p)EU(\pi') \geq p'_{i_0}(\pi'_{i_0}A_{i_0} + \sum_{i \neq i_0} \pi'_i \alpha_i) + (1 - p'_{i_0})\alpha$$

$$\forall i_0 \in \{1, \dots, n\}$$

Hence, for any convex combination  $\tilde{\pi} = \rho\pi + (1 - \rho)\pi', \rho \in (0, 1)$  the following holds:

$$pA + (1 - p)EU(\tilde{\pi}) \geq \tilde{p}_{i_0}(\tilde{\pi}_{i_0}A_{i_0} + \sum_{i \neq i_0} \tilde{\pi}_i \alpha_i) + (1 - \tilde{p}_{i_0})\alpha$$

$$\forall i_0 \in \{1, \dots, n\}.$$

Which shows that  $\succsim_l$  is convex.

mechanism. In contrast, I showed that in the case in which agents need to acquire information on their types, not such trading mechanisms need not even implement the second best learning tree and allocation function. In the case of endogenous information acquisition, a strong-handed mechanism designer might be best. Free trade might simply not pose the correct incentives for information acquisition.

I showed that first-best learning tree is such that agents investigate houses in sequence of their option values (Lemma 1). I complemented this result on the optimality of sequential learning with an example in which the second-best can only be achieved through simultaneous learning (Section 8). Such examples can exist, since simultaneous learning increases the scope of randomness, which is, in turn, a powerful instrument to incentivize learning, as I argued in Section 7.

To derive a characterization of all housing problems in which the first-best is implementable I developed a dynamic version of the revelation principle: I showed that some learning tree and allocation function are implementable if and only if they are truthfully implementable by a mechanism in which the designer asks agents for their types in the sequences mandated by the learning tree to be implemented (Section 5).

It is hoped that further development of the theory would come up with mechanisms that are robust to changes in the designer's and the agent's beliefs about the distribution of values and the learning technology. Of course, starting with the results developed in the present paper, we already know that there are no mechanisms that are ex ante welfare-optimal for any such specification. We might, however, be able to characterize a set of mechanisms that yield ex ante Pareto optima for some large set of assumptions on the distribution of values and the learning technology.

## 10 Appendix

### 10.1 Proof of Lemma 1

Lemma 1 in Section 4 characterizes the first-best learning tree and allocation function  $(l^*, F^*)$ . The present subsection gives a detailed proof via backwards

induction. To be fully general I drop the assumption that no two houses share the same option value, which implies that there might be multiple indexations by option value.

Some more notation is needed to define an object to which backwards induction can be applied. A sequence of values  $\tilde{\omega} = (\omega_{i_1}^1, \omega_{i_2}^2, \dots, \omega_{i_k}^k)$  for  $1 \leq k < n$  is called a **learning history**, with the additional assumption that  $\emptyset$  denotes the initial history before any investigations.<sup>19</sup> Observe that the set of all learning histories  $\tilde{\Omega}$  is naturally embedded in the set of all a posteriori preference profiles  $\bar{\Omega}$ , by identifying the learning history  $\tilde{\omega} = (\omega_{i_1}^1, \omega_{i_2}^2, \dots, \omega_{i_k}^k)$  with the profile of a posteriori preferences  $\bar{\omega}$  according to which  $\bar{\omega}_{i_l}^l = \omega_{i_l}^l$  holds for all  $1 \leq l \leq k$ , additionally the learning history  $\emptyset$  is identified with the a priori profile of preference. The index  $k$  is called the **length of learning history**  $\tilde{\omega} = (\omega_{i_1}^1, \omega_{i_2}^2, \dots, \omega_{i_k}^k)$ . A learning history  $\tilde{\omega}'$  is called a **continuation** of some learning history  $\tilde{\omega}$  of length  $k$  if  $\tilde{\omega}' = (\tilde{\omega}, \omega_{i_{k+1}}^{k+1}, \dots, \omega_{i_l}^l)$  for some  $k < l < n$ .

A **learning subtree**  $l_s$  is a subtree of a learning tree  $l$ . The set of all learning subtrees which, of course, contains the set of learning trees, is denoted by  $L$ . A **learning process** is a function that maps learning histories to (feasible and consistent) continuations of learning. Formally, we have that  $\tilde{l}$  is a function  $\tilde{l} : \tilde{\Omega} \rightarrow L$  with the following two properties. The function is feasible meaning that  $\tilde{l}(\tilde{\omega}) = l_s$  implies the existence of a learning tree  $l$ , in which the learning history  $\tilde{\omega}$  is followed by the subtree  $l_s$ . The function is consistent in the following sense: for any continuation  $\tilde{\omega}'$  of a learning history  $\tilde{\omega}$ , the learning subtree  $\tilde{l}(\tilde{\omega}')$  is determined as the appropriate subtree of  $\tilde{l}(\tilde{\omega})$ . This latter requirement implies that a learning process cannot demand that the first two agents should investigate  $h_1$  and  $h_2$ , while at the same time requiring that agent 2 should investigate  $h_3$  after agent 1 observed that  $h_1$  was of value.

The difference between learning trees and processes lies in the fact that learning processes prescribe learning decisions for every possible learning history, whereas a learning tree only prescribes learning decisions for the learning histories that are reachable via the tree. As an example, consider a learning

---

<sup>19</sup>In Section 4, I restricted attention to learning histories in which agents move in order of their index. For this reason, I only consider such learning histories in the proof.

process  $\tilde{l}$  that prescribes  $h_1$  to be the first house to be investigated. According to this learning process, only two learning histories of length 1 might occur; the first agent might find the house of high or low value. Still  $\tilde{l}$  also prescribes how learning is continued after the learning history, according to which the first agent found  $h_3$  to be of high value. A learning process  $\tilde{l}$  is said to **result in** a learning tree  $l$  if this learning tree arises when all agents follow the learning process, formally,  $\tilde{l}(\emptyset) = l$ . Finally,  $\tilde{F} : \overline{\Omega} \rightarrow M$  denotes an allocation function which is defined on the entire set of a posteriori preference profiles  $\overline{\Omega}$ . The welfare  $\tilde{W}$  of a learning processes  $\tilde{l}$  and allocation functions  $\tilde{F}$  is defined as follows:

$$\tilde{W}(\tilde{l}, \tilde{F}) := W(\tilde{l}(\emptyset), \tilde{F}|_{\tilde{l}(\emptyset)(\Omega)}).$$

The strategy to prove that  $(l^*, F^*)$  maximizes  $W(l, F)$  is to construct a learning process and an allocation function  $(\tilde{l}^*, \tilde{F}^*)$  that satisfy the following two properties:

- (A)  $\tilde{l}^*(\emptyset) = l^*$  as well as  $\tilde{F}^*|_{l^*(\emptyset)(\Omega)} = F^*$
- (B)  $\tilde{W}(\tilde{l}^*, \tilde{F}^*) \geq \tilde{W}(\tilde{l}, \tilde{F})$  for all  $(\tilde{l}, \tilde{F})$

The two claims together imply  $W(l^*, F^*) \geq W(l, F)$  for all  $(l, F)$ , as one can find a  $(\tilde{l}, \tilde{F})$  for any  $(l, F)$ , such that  $\tilde{l}(\emptyset) = l$  as well as  $\tilde{F}|_{l(\emptyset)(\Omega)} = F$ . The upshot of proving this more far reaching hypothesis is that backward induction can be used to show the optimality of  $(\tilde{l}^*, \tilde{F}^*)$ . But to start,  $(\tilde{l}^*, \tilde{F}^*)$  first needs to be defined.

The learning process  $\tilde{l}^*$  is defined such that the first house to be investigated according to learning subtree  $\tilde{l}^*(\bar{\omega})$  has the minimal index among the houses that have not yet been found to be of high value. The allocation function  $\tilde{F}^* : \overline{\Omega} \rightarrow M$  is defined as follows: for every house  $h_i$ , let  $N_i(\bar{\omega})$  be the set of all  $j$  that investigated  $h_i$  and found it to be of value  $A_i$ ; formally,  $N_i := \{j : \bar{\omega}_i^j = A_i\}$ . For all  $i$  with  $N_i(\bar{\omega}) \neq \emptyset$ , let  $\tilde{F}^*(\bar{\omega})(h_i) = \min\{j \in N_i(\bar{\omega})\}$ . Note that these are feasible assignments, since the condition that each agent can investigate at most one house implies that the sets  $N_i(\bar{\omega}) \cap N_{i'}(\bar{\omega})$  for  $i \neq i'$ . Now take the agent with the lowest index who has not yet received a house in the first step and assign him any

house he did not investigate. Repeat this step with the reminder until no houses and agents remain. Observe that such an assignment is always possible, since, on the one hand, no agent can investigate more than one house and, on the other hand, the agent  $n$  cannot investigate any house.

The definitions of  $l^*$ ,  $F^*$ ,  $\tilde{l}^*$ , and  $\tilde{F}^*$  directly imply Claim (A). The proof of Claim (B) needs some work.

To see the optimality of  $\tilde{F}^*$ , observe that for any a posteriori preference profile  $\bar{\omega}$  and allocation  $\mu$  the welfare  $\sum_{i=1,\dots,n} \bar{\omega}_i^{\mu(i)}$  can not be higher than  $\sum_{i=1,\dots,n} \max_{j=1,\dots,n} \bar{\omega}_i^j$ , the sum of values of houses in which each house enters at the maximal value it has for some agent according to  $\bar{\omega}$ . According to  $\tilde{F}^*$ , each house *does* enter this sum at its maximal value: a house  $h_i$  with  $\bar{\omega}_i^j = A_i$  for some  $j$  enters the sum at the highest possible value it can achieve ( $A_i$ ). For all other houses, it holds that  $\max_{j=1,\dots,n} \bar{\omega}_i^j = \alpha_i$ . The allocation function  $\tilde{F}^*$  assigns each of these houses to an agent who does not know its value.

To prove the claim that  $(\tilde{l}^*, \tilde{F}^*)$  maximizes the expected welfare for every learning history, yet more notation is needed. The expected welfare of all agents following the initial history  $\tilde{\omega} \in \tilde{\Omega}$  and then  $\tilde{l}$ , where allocations are determined using  $\tilde{F}^*$ , is defined as  $W(\tilde{\omega}, \tilde{l}, \tilde{F}^*)$ . Note that in this definition we do not consider different allocation functions, since we already established above that there is no other allocation function that improves upon  $\tilde{F}^*$ . Always denote the house with the lowest index that has not yet been found to be of high value at  $\tilde{\omega}$  as  $h_{i^*}$ . The sets of agents and houses still awaiting an assignment at the learning history  $\lambda$ , given that the allocation function is  $\tilde{F}^*$  are denoted by  $N(\tilde{\omega})$  and  $H(\tilde{\omega})$ , respectively. So,  $h_i \notin H(\tilde{\omega})$  if one component of  $\tilde{\omega}$  is  $A_i$ . Let  $W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}, \tilde{F}^*)$  be the expected welfare for the agents in  $N(\tilde{\omega})$  given the learning history  $\tilde{\omega}$  the learning process  $\tilde{l}$  and the allocation function  $\tilde{F}^*$ . So we have that

$$W(\tilde{\omega}, \tilde{l}, \tilde{F}^*) = \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}, \tilde{F}^*)$$

We use induction over the number of remaining learners to show that

$$W(\tilde{\omega}, \tilde{l}^*, \tilde{F}^*) \geq W(\tilde{\omega}, \tilde{l}, \tilde{F}^*)$$

holds for all learning processes  $\tilde{l}$  and all learning histories  $\tilde{\omega}$ . The inductive proof only considers learning processes  $\tilde{l}$  that prescribe to learn houses in  $H(\tilde{\omega})$  at  $\tilde{\omega}$ , since learning a house not in  $H(\tilde{\omega})$  does not increase expected welfare (such a house already enters the welfare sum at its highest possible value).

*Start of Induction:*

Let there be one learner remaining; i.e., let the length of  $\tilde{\omega}$  be  $n - 2$ . The welfare of a learning process  $\tilde{l}$  that prescribes learning  $h_{i'} \in H(\tilde{\omega})$  at  $\tilde{\omega}$  can be calculated as:

$$\begin{aligned}
W(\tilde{\omega}, \tilde{l}, \tilde{F}^*) &= \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}, \tilde{F}^*) \\
&= \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + \sum_{i \in H(\tilde{\omega}), i \neq i'} \alpha_i + p_{i'} A_{i'} + (1 - p_{i'}) \alpha_{i'} \\
&= \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + \sum_{i \in H(\tilde{\omega})} \alpha_i + p_{i'} (A_{i'} - \alpha_{i'}) \\
&\leq \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + \sum_{i \in H(\tilde{\omega})} \alpha_i + p_{i^*} (A_{i^*} - \alpha_{i^*}) = W(\tilde{\omega}, \tilde{l}^*, \tilde{F}^*)
\end{aligned}$$

since  $h_{i^*}$  is the house with maximal option value in  $H(\tilde{\omega})$ .

*Induction Step*

Assume that the hypothesis holds for all learning histories of length  $\geq k$ . Take a learning history  $\tilde{\omega}$  of length  $k - 1$  and assume that the welfare-optimal learning process  $\tilde{l}'$  differs from  $\tilde{l}^*$ . Since we know already from the induction hypothesis that  $\tilde{l}^*$  is optimal for learning histories that have at least length  $k$ , it must be that  $\tilde{l}'$  differs from  $\tilde{l}^*$  in the house that is investigated directly after  $\tilde{\omega}$ , i.e., in stage  $k$ . Assume that  $\tilde{l}'$  prescribes that  $h_{i'} \neq h_{i^*}$  is being

learned after  $\bar{\omega}$ . An upper bound on  $W_{N(\bar{\omega})}(\tilde{\omega}, \tilde{l}', \tilde{F}^*)$  is calculated as follows:

$$\begin{aligned}
& W_{N(\bar{\omega})}(\tilde{\omega}, \tilde{l}', \tilde{F}^*) \\
&= p_{i'} A_{i'} + p_{i'} W_{N((\tilde{\omega}, A_{i'}))}((\tilde{\omega}, A_{i'}), \tilde{l}^*, \tilde{F}^*) + (1 - p_{i'}) W_{N((\tilde{\omega}, a_{i'}))}((\tilde{\omega}, a_{i'}), \tilde{l}^*, \tilde{F}^*) \\
&= p_{i'} A_{i'} + p_{i'} (p_{i^*} A_{i^*} + p_{i^*} W_{N((\tilde{\omega}, A_{i'}, A_{i^*}))}((\tilde{\omega}, A_{i'}, A_{i^*}), \tilde{l}^*, \tilde{F}^*) \\
&\quad + (1 - p_{i^*}) W_{N((\tilde{\omega}, A_{i'}, a_{i^*}))}((\tilde{\omega}, A_{i'}, a_{i^*}), \tilde{l}^*, \tilde{F}^*)) \\
&\quad + (1 - p_{i'}) (p_{i^*} A_{i^*} + p_{i^*} W_{N((\tilde{\omega}, a_{i'}, A_{i^*}))}((\tilde{\omega}, a_{i'}, A_{i^*}), \tilde{l}^*, \tilde{F}^*) \\
&\quad + (1 - p_{i^*}) W_{N((\tilde{\omega}, a_{i'}, a_{i^*}))}((\tilde{\omega}, a_{i'}, a_{i^*}), \tilde{l}^*, \tilde{F}^*)) \\
&= p_{i^*} A_{i^*} + p_{i^*} (p_{i'} A_{i'} + p_{i'} W_{N((\tilde{\omega}, A_{i^*}, A_{i'}))}((\tilde{\omega}, A_{i^*}, A_{i'}), \tilde{l}^*, \tilde{F}^*) \\
&\quad + (1 - p_{i'}) W_{N((\tilde{\omega}, A_{i^*}, a_{i'}))}((\tilde{\omega}, A_{i^*}, a_{i'}), \tilde{l}^*, \tilde{F}^*)) \\
&\quad + (1 - p_{i^*}) (p_{i'} A_{i'} + p_{i'} W_{N((\tilde{\omega}, a_{i^*}, A_{i'}))}((\tilde{\omega}, a_{i^*}, A_{i'}), \tilde{l}^*, \tilde{F}^*) \\
&\quad + (1 - p_{i'}) W_{N((\tilde{\omega}, a_{i^*}, a_{i'}))}((\tilde{\omega}, a_{i^*}, a_{i'}), \tilde{l}^*, \tilde{F}^*)) \\
&\leq p_{i^*} A_{i^*} + p_{i^*} W_{N((\lambda, A_{i^*}))}((\lambda, A_{i^*}), \tilde{l}^*, \tilde{F}^*) \\
&\quad + (1 - p_{i^*}) W_{N((\lambda, a_{i^*}))}((\lambda, a_{i^*}), \tilde{l}^*, \tilde{F}^*) = W_{N(\lambda)}(\lambda, \tilde{l}^*, \tilde{F}^*)
\end{aligned}$$

The first equality follows from the learning process  $\tilde{l}'$  mandating that  $h_{i'}$  be learned after  $\tilde{\omega}$  and the observation that for any learning history of length  $k$ , which includes  $(\tilde{\omega}, A_{i'})$  as well as  $(\tilde{\omega}, a_{i'})$ , we have by induction hypothesis that  $\tilde{l}^*$  is the optimal learning process. The second equality follows from  $\tilde{l}^*$  mandating that  $h_{i^*}$  is being learned next, since it is the house with lowest index in  $H((\tilde{\omega}, A_{i'}))$  as well as in  $H((\tilde{\omega}, a_{i'}))$ . The third equality is owed to a rearrangement of terms together with the observation that an exchange of the order in which houses  $h_{i'}$  and  $h_{i^*}$  are being learned in the preceding history does not change the expected welfare in any of the four relevant cases. The inequality follows from the assumption that the hypothesis of the induction holds for learning histories of length  $k$ . Finally, the last equality follows from the definition of  $i^*$  and  $\tilde{l}^*$ .

This implies that

$$\begin{aligned}
W_{N(\tilde{\omega})}(\tilde{l}^*(\tilde{\omega}), F^*) &= \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}^*, \tilde{F}^*) \\
&\geq \left[ \sum_{i: h_i \notin H(\tilde{\omega})} A_i \right] + W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}, \tilde{F}^*) \\
&= W_{N(\tilde{\omega})}(\tilde{l}(\tilde{\omega}), F^*)
\end{aligned}$$

holds for learning histories of length  $k - 1$ , which concludes the inductive proof of the optimality of the learning process and allocation function  $\tilde{l}^*, \tilde{F}^*$ .

#### *Remarks on Uniqueness*

Any allocation function such that  $\omega_i^{\mu(i)} = \max_{j=1, \dots, n} \bar{\omega}_i^j$  is optimal. The optimal learning process  $\tilde{l}^*$  is unique up to the indexation of houses according to their option value. Said differently: there are as many optimal learning processes as there are indexations of  $H$  that have the feature of  $i \leq i'$  implying that  $O_i \leq O_{i'}$ . Since the indexation in the proof was arbitrary, we know already that all of the resulting learning processes are welfare-optimal. Observe that the crucial inequalities in the start of the induction as well as the inductive step are strict if there is exactly one house in  $H(\tilde{\omega})$  with maximal option value. At the start of the induction, we then have that  $p_{i^*}(A_{i^*} - \alpha_{i^*}) > p_{i'}(A_{i'} - \alpha_{i'})$  for all  $i' \neq i^*$  with  $h_{i'} \in H(\tilde{\omega})$ . Similarly, at the step of the induction we have

$$\begin{aligned}
&p_{i'} A_{i'} + p_{i'} W_{N((\tilde{\omega}, a_{i^*}, A_{i'}))}((\tilde{\omega}, a_{i^*}, A_{i'}), \tilde{l}^*, \tilde{F}^*) \\
&+ (1 - p_{i'}) W_{N((\tilde{\omega}, a_{i^*}, a_{i'}))}((\tilde{\omega}, a_{i^*}, a_{i'}), \tilde{l}^*, \tilde{F}^*) < W_{N((\tilde{\omega}, a_{i^*}))}((\tilde{\omega}, a_{i^*}), \tilde{l}^*, \tilde{F}^*)
\end{aligned}$$

for all  $i' \neq i^*$  with  $h_{i'} \in H(\tilde{\omega})$ . So we can conclude that any learning process that prescribes for houses to be learned in an order that is not consistent with their option value order is suboptimal. This implies, in particular, that there is a unique optimal learning process, and thereby an optimal learning tree if all houses have different option values.

## 10.2 The Problem with $n$ learners

Observe that the above proof makes use of the assumption that only  $n - 1$  agents can investigate houses in the definition of  $\tilde{F}^*$ , the optimal allocation function. That function assigns every house that was found to be of high value to some agent who did find it of high value; and no house that was found to be of low value to an agent who did find this house of low value. If  $n$  agents can investigate houses, such allocations might not be feasible. To see this, consider a housing problem with three houses and assume that the first two agents did find houses  $h_1$  and  $h_2$  of high value and agent 3 found  $h_3$  to be of low value. Clearly, assigning every house that was found to be of high value to agents who did find these houses of high value (houses  $h_1$  to agent 1 and  $h_2$  to agent 2) conflicts with avoiding to assign any house at low value (not assigning  $h_3$  to agent 3). For such cases, the welfare-maximal allocation depends on the specific parameters  $A_i$  and  $a_i$ . To show that this observation not only impacts the calculation of the first-best allocation function, but also the first-best learning tree, I provide the following example.

**Example 4** Let there be three houses  $(p_1, A_1, a_1) = (\frac{1}{2}, 400, 0)$  and  $(p_2, A_2, a_2) = (\frac{1}{4}, 399, -129)$ ,  $(p_3, A_3, a_3) = (\frac{1}{1000}, 98000, 0)$ . Observe that  $O_1 = 100$ ,  $O_2 = 99$ ,  $O_3 \approx 98$ ,  $\alpha_1 = 3$ ,  $\alpha_2 = \frac{77}{4}$ , and  $\alpha_3 = 98$

I use backwards induction to show that the first-best learning tree for the present example differs from the first-best learning tree  $l^*$  as it was described in Lemma 1. Let  $(\tilde{l}^?, \tilde{F}^?)$  be the first-best learning process and allocation function in the given problem. To calculate  $\tilde{F}^?$ , observe that as long as it is feasible to follow the prescription of  $\tilde{F}^*$ , as defined in the proof above, it is optimal to do so. If, however, the first two agents found  $h_1$  to be of low value, and agent 3 found  $h_3$  to be of high value,  $\tilde{F}^*$  does not determine the welfare-maximal allocation. In this case, we have the choice between allocating all houses at their expected values, yielding an expected welfare of  $3 + \frac{77}{4} + 98$ , or allocating  $h_1$  to one of the agents who do not like it and  $h_3$  to agent 3, yielding  $0 + \frac{77}{4} + 98000$ , which is higher. I show next that for the learning history  $\tilde{\omega} = (a_1, a_1)$ , the first-best  $(\tilde{l}^?, \tilde{F}^?)$  prescribe that agent 3 should investigate  $h_3$  and that  $h_3$  should be assigned to 3 if he finds it of high value. The expected welfare difference between this prescription and the prescription of assigning

all houses at their expected value is  $p_3(A_3 + a_1 - \alpha_3 - \alpha_1) > 0$ . Conversely, the expected welfare difference between a prescription, according to which houses  $h_3$  and  $h_2$  exchange roles, and assigning all houses at expected value is  $p_2(A_2 + a_1 - \alpha_2 - \alpha_1)$ . The difference between these two expressions is positive, so learning  $h_3$  implies a larger gain in expected welfare, and we have that  $\tilde{l}^?( \tilde{\omega} )$  prescribes for agent 3 to investigate  $h_3$ . So, which house should be investigated at  $\tilde{\omega}' = (a_1)$ ? The difference between the welfare associated with the second agent learning  $h_1$  or  $h_2$  can be calculated as follows.

$$\begin{aligned} & p_1(A_1 - \alpha_1) + p_1p_2(A_2 - \alpha_2) + (1 - p_1)p_3(A_3 + a_1 - \alpha_1 - \alpha_3) - \\ & p_2(A_2 - \alpha_2) + p_2p_3(A_3 + a_1 - \alpha_1 - \alpha_3) + (1 - p_2)p_1(A_1 - \alpha_1) = \\ p_2p_1(A_1 - \alpha_1) - (1 - p_1)p_2(A_2 - \alpha_2) + (1 - p_1 - p_2)p_3(A_3 + a_1 - \alpha_1 - \alpha_3) = \\ & \frac{1}{4}100 - \frac{1}{2}99 + \frac{1}{4}\frac{1}{1000}(98000 + 0 - 200 - 98) = -\frac{298}{4000} < 0 \end{aligned}$$

Therefore  $\tilde{l}^?( \tilde{\omega}' )$  does not prescribe for agent 2 to investigate  $h_1$ . This yields the conclusion that the first-best learning tree for the given example is not  $l^*$ . To see this, observe that the first-best learning tree is  $\tilde{l}^?( \emptyset )$ . If  $\tilde{l}^?( \emptyset )$  prescribes for agent 1 to initiate learning with an investigation of  $h_1$ , then  $\tilde{l}^?( \emptyset )$  does not prescribe for agent 2 to investigate  $h_1$  again if agent 1 found it to be of high value, which is a difference from  $l^*$ , which would prescribe exactly that. If, on the other hand,  $\tilde{l}^?( \emptyset )$  prescribes for agent 1 to initiate learning with an investigation of a different house, we also established that  $\tilde{l}^?( \emptyset ) \neq l^*$ , as the latter requires for agent 1 to investigate  $h_1$ .

The reason why the characterization of the first-best learning tree and allocation function given in Lemma 1 fails in the case with  $n$  learners is that the definition of the option value no longer applies to this case. The option value was defined to capture the expected welfare increase from learning a house. In the case with  $n - 1$  learners, the option value of house  $h_i$  turned out to be  $p_i(A_i - \alpha_i)$ , reflecting the fact that with probability  $p_i$  the house would be found of high value and in this case the mechanism designer would be able to allocate this house at a utility value of  $A_i - \alpha_i$  above the case when no one would find this house to be of high value.

To see that this reasoning does not apply to the case with  $n$  learners, consider the learning decision for the second agent after the first found  $h_1$  to

be of low value. The relevant option value of learning  $h_1$  is not  $p_1(A_1 - \alpha_1)$ , as in the case of some agent finding  $h_1$  to be of low value, the house  $h_1$  might end up being assigned to an agent who evaluates it at low value. This will happen if the third agent finds  $h_3$  of high value. Therefore, in the case with  $n$  learners, option values cannot be determined independently of the housing problems that they belong to.

## References

- Atila Abdulkadiroglu and Tayfun Sonmez. Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems. *Econometrica*, 66(3):689–701, May 1998.
- A. Bergemann and J. Valimaki. Information in Mechanism Design. In Whitney Newey Richard Blundell and Torsten Persson, editors, *Proceedings of the 9th World Congress of the Econometric Societ*, pages 186–221. Cambridge University Press, 2007.
- O. Compte and P. Jehiel. Information acquisition in Auctions: Sealed-bid or Dynamic Formats? *Rand Journal of Economics*, 38-2:355–372, 2007.
- T.V. Dang. Information Acquisition, Noise Trading, and Speculation in Double Auction Markets. *Discussion Paper, Yale University*, 2008.
- Dino Gerardi and Leeat Yariv. Information Acquisition in Committees. *Games and Economic Behavior*, 62:436–459, 2008.
- A. Gershkov and B. Szentes. Optimal voting schemes with costly information aquisition. *Journal of Economic Theory*, 144, Issue 1:36–68, 2009.
- S. J. Grossman and J.E. Stiglitz. On the Impossibility of Informationally Efficient Markets. *American Economic Review*, 70, Issue 70:393–408, 1980.
- Roger Myerson. Optimal coordination mechanisms in generalized principal-agent problems. *Journal of Mathematical Economics*, 10:67–81, 1982.

- Roger Myerson. Bayesian equilibrium and incentive compatibility: An introduction. In D. Schmeidler L. Hurwicz and H. Sonnenschein, editors, *Social Goals and Social Organization*, Cambridge, 1985. Cambridge University Press.
- Szilvia Papai. Strategyproof Assignment by Hierarchical Exchange. *Econometrica*, 68(6):1403–1433, November 2000.
- Lloyd Shapley and Herbert Scarf. On cores and indivisibility. *Journal of Mathematical Economics*, 1:23–37, 1974.
- Rann Smorodinsky and Moshe Tennenholtz. Overcoming Free Riding in Multi-Party Computations - The Anonymous Case. *Games and Economic Behavior*, 55, Issue 2:385–406, May 2006.
- Robert Verrecchia. Information Acquisition in a Noisy Rational Expectations Economy. *Econometrica*, 50, Issue 6:1415–1430, Nov 1982.