Matching Allocation Problems with Endogenous Information Acquisition

Sophie Bade

September 21, 2010

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, there might not exist any mechanism that implements welfare maximal learning and allocations. Moreover, it might not even be possible to implement the second best through free trade. The paper highlights the value two instruments: randomness and control, in setting incentives for efficient learning. These instruments do not play any role in matching allocation problems with exogenous information.

1 Introduction

The central question of allocation theory is how goods and services should be assigned to people. The answer usually 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. In a market system, the presumption is, by and large, that people’s willingness to buy at a given price communicates enough about their information to enable efficient implementation. However, many allocation problems do not involve markets. Examples are situations in which kidneys, school slots or medical residency programmes need to be allocated to agents. Moreover, in many cases, the requisite information cannot be taken as given but must be acquired at a cost.

In this case mechanisms serve two purposes. On the one hand, they determine allocations for given preferences. On the other hand, they have an impact on the agents’ incentives to acquire information on their preference over the objects that need not be assigned. Mechanisms should then be judged in terms of their allocative properties and by their potential to direct the agents’ learning.

In this paper I study a stylized matching problem in which some objects, henceforth called houses, need to be assigned to some agents. Each agent’s valuation of a particular house is drawn from the same binary distribution. A priori agents are identical in the sense that they do not know their idiosyncratic valuation of any house but only the commonly shared expected values. Learning in the model is specified such that each agent can investigate at most one house, and thereby learn his private valuation of that house. A posteriori preferences are derived from the agents’ strategic learning decisions. In line with the assumption that a designer cannot force an agent to reveal his valuation of house, I also assume that the designer cannot force an agent to investigate a particular house, or to reveal which house he investigated. The goal of the mechanism designer is to maximize overall welfare. These assumptions are stark and stylized. I do not wish to defend them with any particular application in mind. I want to, rather, point out that some general observations on the design of matching mechanisms with endogenous information acquisition can be learned from this simple environment.

The first major observation of the present paper is that welfare optimal behavior might not be implementable in a matching problem with endogenous information acquisition. To establish this result I analytically decompose the agents’ behavior into learning trees and allocation functions. Learn-
ing trees describe the (potentially sequential) learning implied by the agents’ behavior. Allocation functions express allocations as results of the agents’ strategies for given profiles of a posteriori preferences. I show that efficient learning is generally sequential. I go on to characterize the set of all housing problems in which efficient learning trees and allocation functions can be implemented. The fact that there might exist no welfare optimal mechanism for a housing problem with endogenous information acquisition contrasts starkly with the theory of mechanism design for housing problems with exogenous information. If information is exogenously given, then there exists a plethora of mechanisms that implement any Pareto optimum - not just the welfare optima.

The proof of the necessary and sufficient condition for the implementability of the first best entails a range of interesting observations. First of all, since efficient learning is generally sequential, the standard revelation principle fails. It is impossible to ask agents to simultaneously report their (learned) types while at the same time permitting some agents to base their learning decisions on the outcomes of the learning of other agents. Consider a mechanism with an equilibrium strategy profile according to which agent 2 investigates $h_2$ if and only if agent 1 found $h_1$ of high value, otherwise he investigates $h_3$. There is no normal form mechanism that simultaneously elicits types, since agent 2’s type depends on his choice to investigate $h_2$ or $h_3$, which in turn depends on agent 1’s type.

Secondly, I highlight two instruments that are of use in setting incentives for learning: randomness and control. To see the use of randomness consider a problem in which 3 houses $\{h_1, h_2, h_3\}$ need to be assigned to 3 agents. Suppose that the designer hopes to get some agent to investigate $h_1$. Suppose furthermore that the agent would always either investigate $h_2$ or $h_3$ if he is given any of the choices sets $\{h_1, h_2\}$, $\{h_1, h_3\}$ and $\{h_1, h_2, h_3\}$. It might, however, be possible that the agent would choose to learn the value of $h_1$ if he was given a choice between $h_1$ and a non-trivial lottery over $h_2$ and $h_3$, since the attractiveness to investigate $h_2$ or $h_3$ falls with the probability that the agent would obtain this house. For the given problem the designer cannot use serial dictatorship to get the first agent to learn $h_1$, if instead he uses a different mechanism according to which the agent perceives his choices
as one between \( h_1 \) and the lottery over \( h_2 \) and \( h_3 \), the agent will learn \( h_1 \).

A mechanism is said to use the instrument of control if it does not correspond to a mechanism of free trade from some initial ownership structure. I show that the designer can use control to foster learning. From the welfare point of view it might, for example, be wasteful on assign such some house \( h_i \) early one to someone who does not know its value. It might be preferable to wait for some agent to investigate the house at some later stage of the mechanism. Under free trade, it might not be possible to prevent \( h_i \) from being assigned early on: if \( h_i \) has a high expected value some owner of it might decide to keep this house without ever investigating it. In this case control can be used to barr house \( h_i \) in an early stage.

If information about types is exogenous neither one of the two instruments is needed to implement Pareto optimal allocations. If anything, control is a hindrance to efficiency in the case of exogenous information. Conversely, in the case with endogenous information acquisition these instruments can play an important role in setting incentives for efficient learning.

I go on to study the two instruments in isolation. I define the class of direct choice mechanisms in which the designer proposes sets of houses to agents, and agents’ assignments are directly determined by their choices from these sets. Serial dictatorship is an example of a direct choice mechanism: agent 1’s choice out of the grand set of houses is his assignment; agent 2’s choice out of the reminder is his assignment and so forth. In contrast, Gale’s top trading cycles is not a direct choice mechanism. The action of pointing to some house that the agent does not own, does not correspond to the direct appropriation of a house.\(^1\) Direct choice mechanisms can be characterized by the following two features: On the one hand, randomness as an instrument is ruled out. On the other hand, the agents face very simple decision problems in direct choice mechanisms. In fact, the games that are induced by direct choice mechanisms are dominance solvable; to calculate their equilibrium strategies agents only need to be able to determine optimal choices from

\(^1\)Both mechanisms will be defined explicitly in Sections 2.2 and 8.
sets of houses. I provide a straightforward characterization of the class of housing problems in which the first best learning tree and allocation function can be implemented by direct choice mechanisms.

Next I investigate the case in which the designer may not use control. It is well known, that in the case of exogenous information any Pareto optimum can be implemented through mechanisms of free trade. The result that there is generally no mechanism that implements the first best with endogenous information renders the question whether the first best can be implemented through a trading mechanism moot. So I am asking for a yet weaker version of the second welfare theorem: can any second best learning tree and allocation function be implemented through mechanisms of free trade? The answer is again negative: not even the second best needs to be implementable thought trade. Control might be needed to implement the welfare optimal learning tree and allocation function.

The result that the first best learning tree has all agents learn sequentially

---

2To see this consider again the difference between serial dictatorships and Gale’s top trading cycles. An agent does not need to be able to reason strategically to behave optimally under a serial dictatorship: his choice out of a set of houses directly determines his assignment. To see that this is not so for Gale’s top trading cycles mechanism consider the owner of \( h_1 \). To calculate the expected utility associated with the strategy of pointing to some different house \( h_i \) he needs to have a theory on the strategic behavior of the other agents.

3The market failure described here differs markedly from the one described in the literature on information acquisition in markets as exemplified by ?, ? and more recently ?. One of the basic tenets of this literature is that 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 would perfectly aggregate this information, no agent would have any incentive to acquire information. For such incentives to exist, the agents that choose to obtain (costly) information must end up better informed than those that 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. The inefficiency of the present paper is not driven by any issue of free-riding on some other agents’ information acquisition. There is also a major difference in terms of the inefficiency result. ?, ? and ? find that markets are inefficient. Conversely I find, that trading mechanism may not even achieve the second best.
might suggest that sequential learning is always a boon for the designer. I give an example that shows that this is not so. The randomness arising out of simultaneous learning turns out to be a powerful instrument. In fact, in the given example simultaneous learning is required to implement the second best learning tree and allocation function. I use the same example to illustrate the trade-off between efficient learning and efficient allocations when the first best is not implementable.

There is a nascent literature on mechanism design with endogenous information acquisition; a review of this literature can be found in Bergemann and Valimaki (2007). I would like to draw attention to two areas of overlap between the present paper and that literature. One important finding of the present paper is that unconstrained optimal information acquisition is sequential (Theorem 1). Gershkov and Szentes (2009) as well as Smorodinsky and Tennenholtz (2006) present voting models in which the same holds true: in their models the voters’ optimal information acquisition 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. The trade-off between the efficient elicitation of information and the efficient usage of this information has garnered interest in this literature: 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. This trade-off also plays a role in the calculation of optimal allocation mechanisms for housing problems and is featured in Section 9 of the present paper.

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 a priori each agent has the same ranking $h_1 \succ h_2 \succ h_3$ over
the three houses. Assume next 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 $\alpha_3 = -1$. 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 upon 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 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 ac-
quision 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 repre-
sent allocations, where the top (bottom) element represents agent 1’s (3’s)
assignments.

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, in 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 that the game starts with a move of nature, in which the agents actual preferences are determined. I only sketched out subtree following the draw of the state $\omega$. Assume, that this particular $\omega$ is such that every agent values every house highly. Note that to draw the full tree states the initial node $N$ should have branches 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 among investigating house $h_1$ or $h_2$ or not to investigate any house (actions: $l : h_1$, $l : h_2$ and $\emptyset$). All these learning nodes of player one belong to the same information set. Any 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 chose 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 chose, he does
however not know anything about agent 1’s preceding investigation.

To find the unique perfect Bayesian equilibrium observe that $h_1$ is not
agent $j$’s most preferred house if and only if he found $h_1$ 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 chose 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 analytically separate the agents’ learning choices from the choices that matter for the allocation. 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 found $h_1$ to be of high value, otherwise agent 2 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 2.2 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 preferences profiles.

2.3 First Best Learning and Allocations

Serial dictatorship is but one example of a mechanism. To find a mechanism that maximizes ex ante welfare I first establish the learning tree and allocation function that a designer would choose if there were no incentive constraints. Fix a 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 \( h_1 \) the other investigates \( h_2 \). Or one agent investigates \( h_1 \) and the next 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 option. The welfare of three learning trees is

\[
O_1 + O_2 + B, \quad O_1 + (1 - p_1)O_1 + p_1O_2 + B, \quad O_2 + (1 - p_2)O_2 + p_2O_1 + B,
\]

with the following definitions and interpretations: The term \( B = \alpha_1 + \alpha_2 + \alpha_3 \) is a base value of welfare which arises when randomly assigning houses. Next \( O_i = p_i(A_i - a_i) \) is defined as the option value of \( h_i \). The option value of a house can be defined through 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 lean the value of one of these two houses. Intuitively, the option value is a statistic that summarizes the value of learning. In this particular case we have that \( p_1(A_1 - \alpha_1) = 1.5 \) and \( p_2(A_2 - \alpha_2) = 1.25 \), and therefore the optimal learning tree prescribes that agent investigates house \( h_1 \) and agent 2 conditions his investigation on the outcome of that first investigation. This observation is generalized in Theorem 1 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 of 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 immediately 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 2.5
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 $a_2$ corresponds effectively 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 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 $\frac{3}{4}$. Therefore agent 1’s announcement of $a_1$ corresponds to a lottery according to which he obtains $h_2$ with probability $\frac{3}{4}$ and $h_3$ with the complementary probability. Given the choice between $h_1$ and the lottery just described 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 agents.
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.

### 2.6 Two Instruments of Design

The mechanism just described can be viewed as an illustration of two important instruments of design: control and randomness. Loosely speaking a mechanism uses control if it is not strategically equivalent to a mechanism in which the sole role of the designer is to assign property rights and to let people trade. To get some intuition as to why control matters in the present mechanism, suppose there existed a trading mechanism that induced the same learning tree and allocation function. According to this mechanism some agent would have to be the initial owner of $h_1$. This clashes with the fact that according to the mechanism in the example all agents face a positive probability to obtain $h_3$, whereas an owner of $h_1$ would certainly not surrender $h_1$ to obtain $h_3$ given that any agent always values $h_1$ higher than $h_3$ ($a_1 = 0 > -1 = \alpha_3$).

The direct revelation mechanism described above also uses randomness in the sense that agent 1 views his choice as one between a sure outcome, namely $h_1$, and a random outcome, the induced lottery over $h_2$ and $h_3$. Randomness as an instrument is further discussed in Section 3. Observe that agent 1 has to be able to think strategically to infer the lottery that “being waitlisted” is associated with. In Section 4 I discuss a set of mechanisms whose solution does not require such reasoning to determine best responses. These mechanisms rule out the use of randomness as an instrument for designer. I provide a sufficient and necessary condition for the first best learning tree and allocation being implementable by a mechanism that belongs to this class. For now let me claim that both instruments are necessary to implement the first best learning tree and allocation function in the present example. This
will be proved at the end of Section 8.

3 The Model

3.1 Agents, Houses, Values

There are finite sets of agents \( N = \{1, \ldots, n\} \) and of houses \( H = \{h_1, \ldots, h_n\} \). Agent \( j \) values house \( h_i \) at \( \omega^j_i \). The goal of the designer is to maximize welfare by matching houses to agents, formally the objective is to find a bijection \( \mu: H \to N \) such that \( \sum_i \omega^\mu(i)_i \) is maximized. This problem is easily solved, if all values \( \omega^j_i \) are known to the designer. If this is not the case, the question is whether the designer has any means of learning the values \( \omega^j_i \). 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 automatically available 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 do know the values of \( \omega^j_i \). The agents may learn some of their own valuations \( \omega^j_i \).

Now a mechanism not only sets incentives for the allocation of houses for given preferences but also sets incentives for the acquisition of information on houses.

It is assumed that the values of \( \omega^j_i \) are 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. 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.\(^4\)

Formally, an agent’s type is identified with the vector of values he assigns

\(^4\)The significantly more complicated case in which all \( n \) agents can investigate a house is discussed in the Appendix (Section 11.2).
to the houses $\omega^j = (\omega^j_i)_{i=1,\ldots,n}$. The set of all possible types of agent $j$ is denoted by $\Omega^j$. The underlying state space of the model is $\Omega = \Omega^1 \times \cdots \times \Omega^n$, the space of all possible profiles of values $\omega = (\omega^1, \ldots, \omega^n) = (\omega^j_i)_{i,j=1,\ldots,n}$.

The distribution of states, which can be constructed from the assumptions on the distributions of the separate values $\omega^j_i$ is denoted by $P$. The assumption on learning allows me to define the space of a posteriori preference profiles $\overline{\Omega}$ as the space of all $n \times n$ matrices $\overline{\omega}$ with the feature that $\overline{\omega}^j_i = A_i$ or $\overline{\omega}^j_i = a_i$ for at most one $i$ for each $j < n$ and $\overline{\omega}^j_i = \alpha_i$ for all other $i, j$ and the interpretation that $\overline{\omega}^j_i \neq \alpha_i$ holds when house $h_i$ was not investigated by agent $j$.\footnote{This notion of a posteriori preferences corresponds to the moment right after the investigations. Of course, once the agents will 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.}

### 3.2 Allocation Mechanisms

A mechanism $\Gamma = (R, g)$ is a collection of $n$ strategy sets $R := (R^1, \ldots, R^n)$ and an outcome function $g : R \to M$, that maps every strategy profile $r := (r^1, \ldots, r^n)$ to an allocation in $M$. While this definition of a mechanism is standard the function such a mechanism serves in the context of the present setup differs from the standard case. In the standard case agents do know their types $\omega^j$ and do condition their strategies on their types. In the present case types endogenously arise out of learning choices. Said differently: the same mechanism $\Gamma = (R, g)$ induces different games in the case which agents know their types and the alternative case 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 $(T, j, D, I)$ where $T$ denotes the set of nodes, $j$ and $D$ are a functions such that $j(t)$ is the agent who gets to choose from a set of action $D(t)$ at node $t$ and $I$ 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 three subgroups: 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 \( D(t) = \{\emptyset, l : h_1, l : h_2, \ldots, l : h_n\} \) 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. This learning node has three rule nodes \( t_\alpha, t_\beta \) and \( t_\gamma \) for agent 1 as its immediate successors (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.

A share of the information sets \( \mathcal{I} \) is 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, he is also the only one to know the outcome of that investigation.\(^6\)

The terminal nodes specify the payoffs of agents, which are calculated as follows. The decisions in the rules 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. These a posteriori agent \( j \) assigns to house \( h_i \) at some 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 \( \omega_i^t \), where the first branch of the tree leading up to \( t \) is \( \omega \).

\(^6\)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.
Taking again the sketch of a game tree in Figure 2 as an example. Consider the left-most terminal node in the sketch. 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 $s = (s_1, \cdots, s_n)$ and $S := (S_1, \cdots, S_n)$ 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: $g^l : S \times \Omega \to \overline{\Omega}$ and $g^F : S \times g^l(S \times \Omega) \to M$. The function $g^l : S \times \Omega \to \overline{\Omega}$ 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 $\omega^j_i = \omega^j_i$, for all other pairs of agents and houses we have that $\omega^j_i = \alpha_i$. 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 $s$.

At times it is useful to abstract away from particular mechanisms when considering learning and allocations. To do so, 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 $l : \Omega \to \overline{\Omega}$ whose values are determined using a rooted binary tree with house-agent-pairs $(h_i, j)$ as the nodes and

---

7Since 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 $M$. 

18
values $A_i$ and $a_i$ as the edges with the following properties. The root node $(h_i, j_r)$ prescribes agent $j_r$ starts off the learning process with an investigation of $h_i$. 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 $l(\omega) = \omega$ is such that $\omega^i_j = \omega^i_j$ holds for any node $(h_i, j)$ that is visited for $\omega$, otherwise we have $\omega^i_j = \alpha_i$.

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, they 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 that learn houses has no relevance for total welfare. Any learning tree and allocation function that differs only insofar as that agents are moving in a different sequence yields the same welfare. Consequently, 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 an allocation function $F$, through $l(\omega) = g^l(s, \omega)$ and $F(\omega) = g^F(s, \omega)$. For the particular case that the strategy profile under

---

8This 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 an 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:

\[
\max W(l, F) \\
\text{s.t. } \exists \Gamma : \Gamma \text{ implements } (l, F).
\]

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 \(F\).

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

In this section I characterize the learning tree and allocation function \((l^*, F^*)\) 
that maximize social welfare \(W(l, F)\), ignoring implementability. To do so 
it is useful to define the option value of the house as the expected increase 
in welfare due to learning 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 \(O_i := p_i(A_i - \alpha_i)\).
Theorem 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 the house with the highest option value as long as no one found it of high value. Once this happens they should move on to the house with the next highest option value and so forth. Observe that \(F^*\) is well-defined, since for any \(\omega \in l^*(\Omega)\) at most one agent knows a particular house to be of high value. There are generally many optimal learning trees and allocations functions. If there are houses with the same option value, then there are multiple ways to order the houses according to their option value. However, if all houses have different option values then there is a unique efficient learning tree.\(^9\) The set of efficient allocation functions is large, since there are many different ways to assign houses to agents that 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 all histories of investigations. The upshot is that backward induction can be used to characterize the welfare optimal learning process and allocation function \((\tilde{l}^*, \tilde{F}^*)\). The optimal learning process \(\tilde{l}^*\) generalizes \(l^*\) to any possible history of preceding investigations: according to \(\tilde{l}^*\) agents should always investigate the house with the highest option value in the set of houses that have not yet been found to be of high value in any of the preceding investigations. The allocation function \(\tilde{F}^*\) generalizes \(F^*\) insofar as it demands that

\(^9\)Except of course for the reordering of agents, which has been discussed at the end of Section 3.
whenever possible, \( h_i \) should be assigned to someone who evaluates it at \( A_i \), when this is not possible \( h_i \) should be assigned to someone who did not investigate it.\(^\text{10}\) The pair \((\tilde{l}^*, \tilde{F}^*)\) has the feature that it prescribes \((l^*, F^*)\) as the (continuation) learning tree and allocation function after the initial history, where no one has learned anything yet. Consequently \((l^*, F^*)\) is the first best learning tree and allocation function.

To see that \( \tilde{F}^* \) maximizes welfare for any profile of a posteriori preferences observe that a posteriori welfare \( \sum_{i \in N} \omega_i^\mu_i \) is maximized for some allocation function \( \mu \) if each houses \( h_i \), is matched to an agent \( \mu(j) \) who assigns maximal \( \max_j \omega_j^\mu_i \) to this house - conditioning on \( \omega \) being the profile of a posteriori preferences. The function \( \tilde{F}^*(\omega) = \mu \) does precisely that.

To prove the optimality of \( \tilde{l}^* \) by backwards induction consider the \( n-1 \)st learner. Assume that all houses have different option values. Observe that the assumption that there is always an agent who evaluates all houses at their expected value implies that no matter which house the \( n-1 \)st agent learns the designer is free to allocate each house either at its expected value or to some agent who investigated it. 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.

For the inductive step suppose that following \( \tilde{l}^* \) was optimal for agents \( k, \ldots, n_1 \). Should the \( k-1 \)st learner follow \( \tilde{l}^* \)? Suppose not, suppose that according to the welfare optimal learning process \( \tilde{l}^* \) for some history of \( k-2 \) investigations agent \( k-1 \) would have to investigate some house \( h_{\nu} \) which differs from \( h_{\nu^*} \), the house with the highest option value that was not yet found of high value by any agent. Once \( h_{\nu^*} \) has been investigated, only \( k \) learners are left and the induction hypothesis prescribes that the next agent investigates house \( h_{\nu^*} \), no matter the outcome of the investigation of \( h_{\nu^*} \).

Since the investigation in period \( k \) does not depend on what was learned in period \( k-1 \) and we might as well invert the process. The outcome of the

\(^{10}\)The assumption that only \( n-1 \) agents can investigate a house is essential for \( \tilde{F}^* \) to be well-defined, as such assignments might not be feasible with \( n \) learners. To see this, observe that no allocation is consistent with the description of \( \tilde{F}^* \) if all agents found \( h_1 \) to be of low value. This problem renders the extension to the case of \( n \) learners is non-trivial. Since the terminology developed in the proof Theorem 1 turns out to be useful in the discussion of the \( n \)-learner case, this discussion is relegated to the Appendix (Section 11.2).
learning tree according to which agent \( k - 1 \) learns the value of \( h_{i^*} \) and agent
\( k \) learns \( h_{i'} \) no matter what agent \( k - 1 \) found has the same outcome as \( \tilde{\nu} \).
This yields a contradiction as it is possible to construct a learning tree that is associated with higher welfare than this tree: By the hypothesis agent \( k \) has to learn the value of \( h_{i^*} \) if in period \( k - 1 \) the learning agent found \( h_{i^*} \) to be of value \( a_{i^*} \), this is a strict improvement over the supposedly optimal learning tree.

5 Direct Revelation Mechanisms

The original version of the revelation principle as formulated by Myerson
(1982, 1985) does not apply to games of endogenous sequential learning.\(^{11}\) 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 direct 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 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 that 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 agent report their types. Agents are in addition required to obtain the information that they are supposed to truthfully report. If an agent investigates some house \( h_{i'} \) before a rule node \( t \) with \( D(t) = \{A_i, a_i\} \),

\(^{11}\)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.
he cannot possibly truthfully report his type. In that case, he evaluates $h_i$ at $a_i \notin \{A_i, a_i\}$.\footnote{Of course, direct revelation mechanisms could also have been defined using larger sets of announcements $D(t) = \{A_1, a_1, A_2, a_2, \ldots, A_n, a_n\}$. In such mechanisms agents can truthfully report their type after any learning decision. The upshot of “tailoring” direct 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. More on this in footnotes 13 and 15, which refer to the usefulness of the present assumption in the proofs of Theorems 2 and 3 respectively.}

A direct revelation mechanism $\Gamma = (R, g)$ \textbf{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 $g^F(\cdot, s^t) = F$.

\textbf{Example 1} Reconsidering the housing problem defined in Section 2, observe that serial dictatorship is not a direct 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 direct revelation mechanism: First the mechanism designer asks the first agent whether he values $h_1$ highly or not. If the first agent 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 direct revelation mechanism was already discussed in Section 2.5.

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

\textbf{Theorem 2} A learning tree and allocation function $(l, F)$ are implementable by some mechanism $\Gamma$ if and only if they are truthfully implementable by some direct 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 that are 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 $s^e_p$ 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 $s^e_p$ 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 $s^e_p$ in the game induced by $\Gamma'$. Observe that $(l, F)$ is implemented in $\Gamma''$ via the strategy profile that corresponds to $s^e_p$. The reverse conclusion is trivial since direct revelation mechanisms are mechanisms too.\[13\]

Theorem 2 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 simultaneously elicit the agents’ types. 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

\[13\]Referring back to footnote 12 observe that the assumption of binary sets of messages matters for the present proof. If one was to define direct revelation mechanisms with large message sets, as suggested in footnote 12 the present proof would no longer hold. In that case the original strategy profile $s^e$ is silent on how agents should play on the paths following learning choices not prescribed by $s^e$. 25
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

To understand the problems associated with the implementation of the first best learning tree and allocation function, consider a direct 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 direct 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. This last condition implies, for example, that the designer may not promise the same house to two different agents.

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 better grasp the requirement, 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 $\pi := (\pi_1, \ldots, \pi_n)$ 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 $h_i \gtrsim I \pi$.\footnote{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.}

Observe that statements such as $\pi' \gtrsim I h_i$ or $\pi' \gtrsim \pi$ 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 $h_1 \gtrsim I h_2$ amounts to the following inequalities:

\[
A_1 > \alpha_2 > a_1 \\
p_1 A_1 + (1 - p_1) \alpha_2 \geq p_2 A_2 + (1 - p_2) \alpha_1
\]

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 $h_1 \gtrsim I h_2$ 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 $\gtrsim I$ 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, agents are happy to investigate houses in the order prescribed by the first best learning tree. I analyze this case in Theorem 4. The problem lies in the fact that these two relations do not generally coincide. To see this observe that the order by option values is necessarily transitive whereas the restriction $\gtrsim I$ to the set of houses need not be transitive. In the introductory example $h_1 \gtrsim I h_2$ and $h_2 \gtrsim I h_3$, but $h_1 \gtrsim I h_3$ does not. So the two relations cannot generally coincide.

Interestingly, though, there are cases in which the designer can artfully use the strategic behavior of agents to implement the first best even if the two orders do not coincide: The mechanism defined in Section 2.5 implements the first best in the introductory example, even if $h_1$ and $h_3$ are such that $O_1 > O_3$ holds whereas $h_1 \gtrsim I h_3$ does not. Theorem 3 delineates the set of all housing problems in which the first best $(l^*, F^*)$ is implementable.
To state this theorem it is convenient to introduce some more notation for direct revelation mechanisms. Let \( d_A(t) \) as the number of high value statements in the history leading up to \( t \). Define a function \( i : T_R \rightarrow \{1, \ldots , n\} \) 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 makes announcement \( X \) at node \( t \) and given that all other agents follow the strategy profile \( s \). Let \( \pi_i(t, s, X) = (\pi_1(t, s, X), \ldots , \pi_n(t, s, X)) \) stand for the corresponding lottery over houses. I say that the indices on \( H \) reflect an \textbf{option-value-indexation} if \( O_i \geq O_{i'} \) holds for all \( i < i' \).

**Theorem 3** The first best \((l^*, F^*)\) is implementable if and only if there exists a direct revelation mechanism \( \Gamma \) and some option value-indexation on \( H \), 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 \( \pi_i(t, s^t, a_{i(t)}) = 0. \)
- \( h_{i(t)} \succsim \pi (t, s^t, a_{i(t)}). \)

**Proof** By Theorem 2, \((l^*, F^*)\) can be implemented if and only if it can be implemented truthfully through a direct revelation mechanism. Let \( \Gamma = (R, g) \) be the direct 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 \( t \).

Observe that Theorem 3 can be split into a “mechanical” and an “artful” part. The first two conditions straightforwardly determine some properties

---

\[15\] Note that all nodes are reached with positive probability, therefore we need not be concerned with out of equilibrium beliefs. If I had defined direct revelations mechanisms allowing for messages spaces large enough to allow for the announcement of any possible a posteriori type as suggested in footnotes 12 and 13, this would not hold.
of any direct 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 matters with 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 \(l^*\). Theorem 3 has the following immediate corollary:

**Corollary 1** In some housing problems with endogenous information acquisition the first best learning tree and allocation function is not implementable.

**Proof** The following example contains the proof:

**Example 2** Define a housing problem with three agents through 
\[ h_1 = (p_1, A_1, a_1) = (\frac{1}{2}, 70, -10), \]
\[ h_2 = (p_2, A_2, a_2) = (\frac{2}{3}, 100, 0), \]
and \( h_3 = \alpha_3 = -2 \). Observe that \( \alpha_1 = 30, \alpha_2 = 75, O_1 = 20, \) and \( O_2 = \frac{25}{4} \), 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 direct revelation mechanism that implements it according to Theorem 3. 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’s have a look at the node after a declaration of \( A_1 \) by the first agent. We must have that \( \pi(A_1, s^l, A_2) = h_2 \) 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 \( \pi(A_1, s^l, a_2) = h_3 \) as \( h_3 \) is the only other remaining house. However, agent 2 is willing to investigate \( h_2 \) in this node as \( h_2 \not\succ h_3 \) does not hold (since \( a_2 > \alpha_3 \)).

This Corollary stands in stark contrast with the large set of mechanisms that can be used to implement any Pareto optimal allocation in housing problems without endogenous information acquisition. So while it is easy to satisfy the goal of efficient allocations alone, the addition of the goal that information should be acquired in an efficient manner renders the problem quite a bit more complicated. The example demonstrates that there are no mechanisms that generally satisfy both goals. Observe, that the very parsimony
of the deviation from the standard model without endogenous information acquisition strengthens the point being made here. Neither multiple learning decisions, nor correlated values, nor some explicate cost of learning are needed to show that there might not exist any mechanism to implement the first best in a housing problem with endogenous information acquisition.

Note also that the result would not change if we would allow the designer to exogenously randomize. In the presentation of the introductory example I argued that the use of non-degenerate lotteries can facilitate implementation. So the set of implementable learning trees and allocation functions \((l, F)\) generally becomes larger when allowing for exogenous randomization. However in first best \((l^*, F^*)\) in Example 2 does not become implementable when allowing for exogenous lotteries. In fact the arguments provided in Example 2 translate word by word to the larger set of mechanism.

7 Direct Choice Mechanisms

This section is concerned with the particular case of mechanisms in which at any rule node agents get to choose from sets \(D(t) \subset H\) where the action “\(h_i\)” is to be interpreted as the appropriation of \(h_i\). These mechanisms are called \textbf{direct choice mechanisms} since the actions of agents directly translate to choices of houses. Serial dictatorship is an example of a direct choice mechanism. Conversely, the mechanism used to implement \((l^*, F^*)\) in Section 2.5 is not a direct choice mechanism since agent 1’s assignment after an announcement of \(a_1\) also depends on the strategy agent 2.

Direct choice mechanisms are of interest since they present very simple choices for agents: backward induction is not needed to figure out best replies. Agents do not have to be able to think strategically; they only need to be able to solve single agent maximization problems to follow equilibrium behavior. The following example illustrates the implementation of \((l^*, F^*)\) in some housing problem by a direct choice mechanism that is not a serial dictatorship:

\textbf{Example 3} Define a housing problem with four agents through 

\( h_1 = (p_1, A_1, a_1) = (\frac{3}{4}, 8, 0), \ h_2 = (p_2, A_2, a_2) = (\frac{1}{2}, 3, -2), \) and \( h_3 = h_4 = \)
\[ \alpha_3 = \alpha_4 = \frac{1}{4}. \]
Observe that \((l^*, F^*)\) is implementable through a direct choice mechanism which lets each agent choose from the pair of houses with the highest and lowest index among the houses still available.\(^\text{16}\)

Observe that no agent has two rule nodes on any path of a direct choice mechanism (for otherwise the agent’s assignment would not be determined by his choice in the first rule node on that path). In terms of the instruments of design, direct choice mechanisms rule out randomness but do allow for control. In the sequel I only consider direct choice mechanism with binary choice sets, that is with \(D(t) = \{h_{i(t)}, h_{i'(t)}\}\) for any \(t \in T_R\) and some \(i(t), i'(t)\). This is without loss of generality since an agent’s optimal strategy at any rule node \(t \in T_R\) of a direct choice mechanism either prescribes to always choose the same house or to condition the choice on the investigation of some house. In that latter case he chooses the investigated house if he finds it of high value, if not he chooses a fixed consolation alternative.\(^\text{18}\)

No other house is ever chosen at \(t\).

**Theorem 4** The first best learning tree and allocation function \((l^*, F^*)\) are implementable by a direct choice mechanism if and only if \(h_i \succeq h_{i'}\) holds for all \(i < i'\) for some option value indexation of \(H\).

**Proof** Define a direct choice mechanism \(\Gamma\) through the following game tree: For all rule nodes \(t \in T_R\) let \(D(t) := \{h_{i(t)}, h_{i'(t)}\}\) with \(i, i'\) defined inductively as follows. Offer \(\{h_1, h_n\}\) to agent 1. For all ensuing rule nodes \(t\) let \(i(t) = i(t')\) and \(i'(t) = i(t') - 1\), if agent \(j(t')\) chose house \(h_{i'(t')}\) at the prior rule node \(t'\). If \(j(t')\) did choose the other house, let \(i(t) = i(t') + 1\) and \(i'(t) = i(t')\).

---

\(^{16}\)The present example differs from the introductory example only insofar as there are two houses (as opposed to one) of known value in the present example and insofar as the value of these houses is higher than the value of the known house in the introductory example.

\(^{17}\)It is of interest to note that serial dictatorship does not implement \((l^*, F^*)\) in the present example, since the first dictator would never appropriate \(h_3\) or \(h_4\). More on that in Section ??.

\(^{18}\)Remember that the solution concept is pure strategy perfect Bayes Nash equilibrium, which implies that agents cannot mix over indifferent alternatives.
I claim that \( \Gamma \) implements \((l^*, F^*)\) via the equilibrium strategy profile \( s^e \) according to which before node \( t \) agent \( j(t) \) learns the value of house \( h_{i(t)} \) and keeps it if and only if he finds it of high value. To see that \( s^e \) is an equilibrium observe that \( h_{i(t)} \succ_i h_{i'(t)} \) for all \( t \) as \( \succ_i \) coincides with the option value ranking (indexation) and \( i(t) < i'(t) \) for all \( t \). To see that \((l^*, F^*)\) is being implemented by \( \Gamma \), observe that the \( s^e \) always prescribes that the house with the lowest index among the houses that have not yet been found of high value is investigated next. With respect to the allocation function implemented by \( \Gamma \) observe that any house that was found to be of high value by an agent is immediately taken off the market by that agent, all other houses are assigned at their expected values.

Now let \((l^*, F^*)\) be implemented by a direct choice mechanism \( \Gamma \). Suppose that \( h_i \succ_i h_{i'} \) was not valid for some \( i < i' \). Let us focus on the learning history \((A_1, A_2, \ldots, A_{i-1}, a_i, \ldots, a_i)\) which has length \( n - 2 \), starts with high values for the first \( i - 1 \) houses and continues with finding the low value for \( h_i \) for the next \( n - 2 - (i - 1) \) investigations. Since \( \Gamma \) implements \( l^* \), this learning history occurs for some state \( \omega \) according to the equilibrium strategy profile. Observe that the first \( i - 1 \) agents appropriate houses \( h_1, \ldots, h_{i-1} \). Since \( i - 1 < i < i' \), none of them appropriates \( h_i \) or \( h_{i-1} \). None of the next \( n - 2 - (i - 1) \) agents appropriates \( h_i \), as they all find it to be of low value. Now observe that if \( h_{i'} \) would have acted as the consolation alternative in the choice set of any of these agents, the agent could not have investigated \( h_i \) since, by assumption, \( h_i \succ_i h_{i'} \) does not hold. Therefore, no agent \( j \in \{i \cdots n - 2\} \) appropriates \( h_{i'} \). It can be concluded that the choice set of agent \( n - 1 \) must be a subset of \( \{h_i, h_{i'}\} \). Now the prescription of \( l^* \) that \( h_i \) is being investigated at this junction contradicts with the assumption that \( h_i \succ_i h_{i'} \) does not hold. \( \square \)

Theorem 4 shows that the coincidence between the learning relation and the option value ranking is of essence for the implementation of the first best through direct choice mechanisms. The following lemma illuminates the relation between the option value ranking and the learning relation \( \succ_i \).

**Lemma 1** The learning relation \( \succ_i \) on some set of houses \( H \) coincides with the option value ranking expressed by the index of the houses if one of the
three following conditions holds for all \(i \in \{1, \ldots, n\}\).

(A) \(\alpha_i = \alpha\) for some fixed \(\alpha\)

(B) \(A_i > A_{i+1}\) and \(a_{i+1} > a_i\)

(C) \(A_i > a_{i+1} > a_i\) but not \(A_{i+1} > a_i > a_{i+1}\).

Moreover for any two houses \(h_i\) and \(h_{i+1}\) with \(A_i > a_{i+1} > a_i\) one can construct houses \(h(O)\) such that the expected value of \(h(O)\) is \(\alpha_i\), the probability that the good outcome arises is \(p_i\), and the option value is \(O\) (so that \(h(O_i) = h_i\) then there exists a number \(O(h_i, h_{i+1})\) such that \(h(O) \gtrless h_i\) for all \(O > O(h_i, h_{i+1})\).

**Proof** Observe that \(p_i A_i + (1-p_i) \alpha_i - p_i A_i - (1-p_i) \alpha_i\) can be reformulated as \(p_i A_i - p_i \alpha_i - (p_i A_i - p_i \alpha_i) + \alpha_i - \alpha_i\) as well as \((A_i - A_i) p_i p_i + (1 - p_i) (a_i - a_i)\). Under condition (A) the first reformulation reduces to \(O_i - O_i\) which is non-negative for \(i < i'\). Condition (B) implies that the second reformulation is nonnegative for \(i < i'\). Therefore either condition implies \(p_i A_i + (1-p_i) \alpha_i \geq p_i A_i - (1-p_i) \alpha_i\) for \(i < i'\). This, together with the observation that \(a_i < \alpha_i \leq A_i\), holds for all \(i < i'\) in either case, implies that \(h_i \gtrless h_{i'}\) for all \(i < i'\).

To see that condition (C) implies to coincidence of the two rankings observe that the inequality \(A_i > a_{i+1} > a_i\) implies that investigating \(h_i\) is preferred to choosing \(h_i\) unconditionally \((p_i A_i + (1-p_i) \alpha_{i+1} > p_i A_i + (1-p_i) \alpha_i = \alpha_i)\) and to choosing \(h_{i+1}\) unconditionally \((p_i A_i + (1-p_i) \alpha_{i+1} > p_i \alpha_{i+1} + (1 - p_i) \alpha_{i+1} = \alpha_{i+1})\). Conversely, \(A_{i+1} > a_i > a_{i+1}\) not holding implies that either \(\alpha_i\) or \(\alpha_{i+1}\) is an upper bound for \(p_i A_i + (1-p_i) \alpha_i\) the utility of investigating \(h_2\). In sum, we have that \(p_i A_i + (1-p_i) \alpha_i \geq p_i A_i + (1-p_i) \alpha_i\) which implies that \(h_i \gtrless h_{i+1}\).

To see the last claim of the lemma, let \(h(O) = (p_i, \frac{Q}{p_i} + \alpha_i, \alpha_i - \frac{Q}{1-p_i})\). Observe that, \(h(O) \gtrless h_{i+1}\) holds if and only if \(p_i (\frac{Q}{p_i} + \alpha_i) + (1-p_i) \alpha_{i+1} > p_{i+1} A_{i+1} + (1 - p_{i+1}) \alpha_i\) which implies that \(h(O) \gtrless h_{i+1}\) holds for \(O \geq O(h_i, h_{i+1}): = p_{i+1} A_{i+1} + (1-p_{i+1}) \alpha_i - p_i \alpha_i - (1-p_i) \alpha_{i+1}\).\(\square\)

Note that the lemma together with Theorem 4 imply that the first best \((l^*, F^*)\) can be implemented if condition (A), (B) or (C) holds. Condition
(A) describes the case in which all houses share the same expected value
expected value. Condition (B) concerns the case in which the low and high
values of houses are nested. It is important to note that we do not even
have to calculate explicit option values to use this criterion to determine a
direct choice mechanism that implements the first best. If there exists an
indexation of houses such that $|\frac{1}{2} - p_i| > |\frac{1}{2} - p_{i+1}|$, \(A_i > A_{i+1}\) and \(a_{i+1} > a_i\)
for all \(i\), then the option value ranking and the ranking by \(\succ\) coincide, they
are both expressed by the index.\(^{19}\) According to condition (C) the expected
value houses with higher index is intermediate between the high and low
values of houses with lower index without the converse holding. The last
part of the Lemma singles out the case in which houses are similar enou gh in
terms of their expected values while being different enough in terms of their
option values, formally \(a_i < \alpha_{i'} < A_i\) and \(O_i > O(h_i, h_{i'})\) for all \(i < i'\).
Combining Theorem 4 with Lemma 1 we can identify a range of housing
problems in which the first best \((l^*, F^*)\) is implementable through direct
choice mechanisms.

8 Trading Mechanisms

The mechanism constructed in Section 2.5 uses randomness to implement
the first best \((l^*, F^*)\). When agent 1 declares \(a_1\) his assignment depends on
the choice of agent 2. This choice is in turn based on his type and therefore
random from the point of view of agent 1. This randomness is used to level
out the impact of \(h_3\) which itself cannot be used to incentivize the learning
of \(h_1\).

There are two reasons why such randomness facilitates incentivization.

\(^{19}\)Note that if the condition (B) holds then \(|\frac{1}{2} - p_i| > |\frac{1}{2} - p_{i+1}|\), \(\nu(A_i) > \nu(A_{i+1})\) and
\(\nu(a_{i+1}) > \nu(a_i)\) holds for all \(i\) for any strictly increasing function \(\nu\). The learning tree and
allocation function \((l^*, F^*)\) as characterized in Theorem 1 is also first best if the designer
converts utility values with an increasing function \(\nu\). Furthermore by Lemma 1 it can
be implemented through a direct choice mechanism. Summing up we have a particularly
strong result on the robustness of implementation in case (B) of Lemma 1: no matter the
degree of inequality aversion of the designer, the first best learning tree and allocation
function is \((l^*, F^*)\), and it can always be implemented through a direct choice mechanism.
The same reasoning does not apply to conditions (A) and (C).
First of all the attractiveness to investigate a house increases with the probability that the agent would obtain this house if he finds 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 that has the choice between $h_1$ and $h_2$. Conversely, for an agent that 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 $h_i \succsim_l \pi$ together with $h_i \succsim_l \pi'$ implies that $h_i \succsim_l \rho \pi + (1 - \rho) \pi'$.\(^{20}\)

The present section complements the preceding one insofar as that the preceding studied implementation under the assumption that the designer may use control but not randomness as an instrument. The present section asks the complementary question for implementation though mechanisms that allow for randomness but rule out control. Given that a mechanism

\(^{20}\)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

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

and

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

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

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

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

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

Which shows that $\succsim_l$ is convex.
uses no control if it is a free trade mechanism in the sense that the sole role of the designer is to assign property rights to agents and to then let them trade freely, the guiding question of the present section can be reframed as the question whether a version of the Second Fundamental Theorem of Welfare holds for housing problems with endogenous information acquisition.

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) it holds that 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 set of welfare optima (which is of course a small subset of the set of Pareto optima) can be implemented through trade.

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).

Of course, we already know from Corollary 1 that the first best \( (l^*, F^*) \) need not be implementable by any mechanism, not to mention by a mechanism of free trade. In the present section I show that not even the second best \( (l, F) \) needs to be implementable by free trade.

**Theorem 5** The second best learning tree and allocation function \((l^0, F^0)\) might not be implementable through Gale’s top trading cycles mechanism.
For a sketch of the proof reconsider the introductory example. I already showed in Section 2.5 that there exists a mechanism that does implement the first best \((l^*, F^*)\). I next show that this is the unique direct revelation mechanism that implements \((l^*, F^*)\). To see this consider a direct 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 \(\mathcal{ω}\) with \(\mathcal{ω}_1 = A_1\) 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 accord with a welfare 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 \(h_1 \sim_{l} \pi\). 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 direct 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 direct 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. This stands in conflict with the following two observations. Firstly, some agent must start out the trading process as the owner of \(h_1\). Secondly, there are no a posteriori preferences according to which the owner of \(h_1\) would trade in \(h_1\) for \(h_3\). We can conclude that in the introductory housing problem the second best \((l^*, F^*)\) (which is equal to the first best \((l^*, F^*)\)) cannot be implemented through Gale’s top trading cycles mechanism.

As an aside, observe that in that example \(h_1 \gtrsim_l h_3\) does not hold. Using
Theorem 4 we know that there is no direct choice mechanism that implements \((l^*, F^*)\) in that example. So we can conclude that the designer must use both instruments control and randomness to implement the first best \((l^*, F^*)\) in that example. This proves the claim that both instruments: control and randomness, are necessary to implement the first best \((l^*, F^*)\) in the introductory example. (This claim was made at the very end of Section 2.) One might critique 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 5 also holds true for this much larger class of mechanisms. Since the definition of hierarchical exchange mechanism is cumbersome, I relegate the statement and proof of this much stronger result (Theorem 6) to the Appendix. The proof shows that the first best \((l^*, F^*)\) in the housing problem defined in Example 3 cannot be implemented through a hierarchical exchange mechanism. Given that I already constructed a direct choice mechanism that does implement the first best \((l^*, F^*)\) in that housing problem, the example shows at the same time that the instruments of randomness and control are not substitutes. The proof of Theorem 5 immediately follows from the proof of Theorem 6 and is, therefore, omitted here.

9 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 \((l^o, F^o)\) 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. In the following example I 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 \((l^o, F^o)\) 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 \((l^o, F^o)\).

**Example 4** Reconsider Example 2. Define a direct 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 \(g'(\cdot, s^t) = l^*\). 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 \( h_2 \succ h_1 \) 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 is facing a choice between \( h_1 \) and \( h_3 \). Since \( h_1 \succ h_3 \) 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: \( \pi(\emptyset, s^t, A_1) = \left( \frac{3}{4}, \frac{1}{4}, 0 \right) \) 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:

\[
\frac{1}{2} \left( \frac{3}{4}70 + \frac{1}{4}(-2) \right) + \frac{1}{2} \left( \frac{1}{2}75 + \frac{1}{2}(-2) \right) - \frac{3}{4} \left( \frac{1}{2}100 + \frac{1}{2}(-2) \right) + \frac{1}{4} \left( \frac{3}{4}30 + \frac{1}{4}(-2) \right) > 0.
\]

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

\[
\begin{array}{ccc}
& h_1 & \text{no thanks} \\
\hline
h_2 & (h_1, h_2, h_3) & (h_1, h_3, h_2) \\
\text{no thanks} & (h_3, h_2, h_1) & (h_3, h_1, h_2)
\end{array}
\]

“\( h_j \)” 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 $s_1$, agent 2 chooses from $\{h_2, (\frac{1}{4}, 0, \frac{1}{2})\}$, where $h_2 \succeq (\frac{1}{4}, 0, \frac{1}{2})$ holds as $\alpha_2 > \frac{1}{5}70 + \frac{1}{7}(-2)$ and $A_2 > E\left(\frac{1}{4}, 0, \frac{1}{2}\right) > a_2$.\footnote{Observe that this is the only strategy profile that survives the iterated elimination of dominated strategies.}

The learning tree $\overline{l} = \overline{g}(\cdot, s^e)$ is such that agent $j$ unconditionally investigates house $h_j$ for $j = 1, 2$ and therefore $\overline{\omega} \in l(\Omega)$ if and only if $\overline{\omega}_i \neq \alpha_i$ for $i = 1, 2$. The allocation function $\overline{g}^F(\cdot, s^e) = \overline{F}$ implemented by $\overline{\Gamma}$ 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 $\overline{F}$ is an efficient allocation function.

To find the second best $(l^0, F^0)$ we do not need to look any further: We already found that $F^r$ is the allocation function that maximizes $W(l^r, F)$ subject to $(l^r, F)$ being implementable. Next observe that $\overline{l}$ differs from $l^r$ only insofar as that $l^r$ prescribes that $h_1$ is investigated after $h_1$ was found of low value whereas $\overline{l}$ prescribes that $h_2$ is investigated in that same case. This is smallest possible deviation from $l^r$. Given that this learning tree $\overline{l}$ is implementable together with an efficient allocation function $\overline{F}$, there cannot possibly be another implementable pair $(\hat{l}, \hat{F})$ such that $W(\hat{l}, \hat{F}) > \max(W(l^r, F^r), W(\overline{l}, \overline{F})).$

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 $W(l^r, F^r)$ and $W(\overline{l}, \overline{F})$ can be expressed as the difference between $p_1p_2(A_1 - \alpha_1) + p_1p_2(A_2 - \alpha_2) + (1 - p_1)p_1(A_1 - \alpha_1) + (\alpha_1 + \alpha_2 + \alpha_3)$ and $p_1(A_1 - \alpha_1) + p_2(A_2 - \alpha_2) + (\alpha_1 + \alpha_2 + \alpha_3)$ which comes down to $O_1(p_2 - p_1) - O_2(1 - p_1) = 20(\frac{3}{4} - \frac{1}{2}) - \frac{25}{7}(1 - \frac{1}{2}) < 0.$

An important observation to carry away is that $h_1$ and $h_2$ are being investigated simultaneously according to the second best $(l^r, F^r)$. 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: it’s 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 they will end up with \( h_3 \) is low. When learning is sequential such belief cannot be generated.

Also note that the calculation of the second best \((l, F)\) 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 \( \tilde{l} \) 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.

10 Conclusion

brainstorm for new 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 there is a plethora of mechanisms that can achieve any desires Pareto optimum in a housing problem. Conversely I showed that in the case in which agents need to acquire information on their types there are generally no mechanisms that achieve first best welfare (Corollary 1).

In the standard case of exogenous information it holds furthermore that all Pareto optima are achievable through trading mechanisms. Of course, the observation first best might not even be achievable through any mechanism, renders the question whether the first best can generally be achieved through free trade idle. So I asked the question whether the second best might always
be achievable through a trading mechanism. Also this question was answered in the negative. 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. 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.

I showed that first best learning tree is such that agents investigate houses in sequence of their option values. 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.

The characterization of all housing problems in which the first best is implementable shed some light on two instruments which are of use in housing problems with endogenous information acquisition: randomness and control. This contrasts starkly the theory of housing problems with exogenous types. In that environment any Pareto optimum can be implemented through serial dictatorship, which is a mechanism that neither involves randomness nor control (at least when interpreting the set of “trading mechanisms” as the set of hierarchical exchange mechanisms, which encompass serial dictatorships). Conversely, I showed that not only is the first best learning tree and allocation function implementable in the housing problem defined in Section 2, its implementation necessarily involves the use of both instruments: randomness and control.

Considering both instruments in isolation I showed that the class of housing problems in which the first best can be implemented by mechanisms that do only allow for the instrument of control and not for randomness can simply be characterized by the coincidence of the learning- and the option value-order on the set of houses. What is more the set of mechanisms which only use control are of some interest in their own right, since they do not
place high demands on the agents’ ability to reason strategically to solve for equilibria. Even agents that are only able to solve single person maximization problems are able to determine equilibrium strategies in direct choice mechanisms.

Finally I discussed some important trade-offs in the design of mechanisms at the hand of an example in which the first best learning tree and allocation function are not implementable. One the one hand the first best learning tree prescribes that houses should be investigated sequentially (at least if they all have different option values). On the other hand simultaneous investigations increase the scope of randomness as perceived by the agents. A second trade-off concerns efficient learning and efficient allocations.

One drawback of the present approach that the designer has to know the distribution of values $P$ and the learning technology to design the optimal mechanism. Depending on the type of mechanism used agents also have to know $P$ to calculate their equilibrium behavior. While in direct choice mechanism agents do not need to know either to determine their dominant strategies, more general mechanism do require that agents base their optimal choices on beliefs about the behavior and types of others. 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 $P$ and and the learning technology.

11 Appendix

11.1 Proof of Theorem 1

Theorem 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.

Some more notation is needed to define an object to which backwards induction can be applied. A sequence of values $\tilde{\omega} = (\omega^1_{i_1}, \omega^2_{i_2}, \ldots, \omega^k_{i_k})$ for $1 \leq k < n$ is called a learning history, with the additional assumption
that $\emptyset$ denotes the initial history before any investigations. Observe that the set of all learning histories $\tilde{\Omega}$ is naturally embedded in the set of all a posteriori preference profiles $\Omega$, by identifying the learning history $\tilde{\omega} = (\omega_{i_1}^1, \omega_{i_2}^2, \ldots, \omega_{i_k}^k)$ with the profile of a posteriori preferences $\omega$ according to which $\omega_{i_l}^l = \omega_{i_l}^l$ 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 $\tilde{\omega}$. A learning history $\tilde{\omega}'$ is called a continuation of some learning history $\tilde{\omega}$ of length $k$ if $\tilde{\omega}' = (\tilde{\omega}, \omega_{i_{k+1}}^{k+1}, \ldots, \omega_{i_l}^l)$ 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 $h_1$ to be of high 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 process $\tilde{l}$ that prescribes that $h_1$ is 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

---

22 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.
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 process $\tilde{l}$ 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 $(\tilde{l}^*, \tilde{F}^*)$ which satisfy the following two properties:

\begin{align*}
(A) & \quad \tilde{l}^*(\emptyset) = l^* \quad \text{as well as} \quad \tilde{F}^*|_{\tilde{l}^*(\emptyset)(\Omega)} = F^* \\
(B) & \quad \tilde{W}(\tilde{l}^*, \tilde{F}^*) \geq \tilde{W}(\tilde{l}, \tilde{F}) \quad \text{for all } (\tilde{l}, \tilde{F})
\end{align*}

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 $\tilde{F}|_{\tilde{l}(\emptyset)(\Omega)} = F$. 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 $\tilde{F}^*(\overline{\omega})$ has the minimal index among the houses that have not yet been found of high value. The allocation function $\tilde{F}^* : \overline{\Omega} \rightarrow M$ is defined as follows: For every house $h_i$ let $N_i(\overline{\omega})$ be the set of all $j$ that investigated $h_i$ and found it to be of value $A_i$; formally $N_i := \{ j : \overline{\omega}_j^i = A_i \}$. For all $i$ with $N_i(\overline{\omega}) \neq \emptyset$ let $\tilde{F}^*(\overline{\omega})(h_i) = \min \{ j \in N_i(\overline{\omega}) \}$. Note that these are feasible assignments since the condition that each agent can investigate at most one house implies that the sets $N_i(\overline{\omega}) \cap N_{i'}(\overline{\omega})$ 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 $\omega$ and allocation $\mu$ the welfare $\sum_{i=1,...,n} \omega_i^{\mu(i)}$ can not be higher than $\sum_{i=1,...,n} \max_{j=1,...,n} \omega_i^j$, the sum of values of houses in which each house enters at the maximal value it has for some agent according to $\omega$. According to $\tilde{F}^*$ each house does enter this sum at its maximal value: A house $h_i$ with $\omega_i^j = A_i$ for some $j$ enters the sum at the highest possible value it can achieve ($A_i$). For all other houses it holds that $\max_{j=1,...,n} \omega_j^i = \alpha_i$.

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 when learning starts off with the initial history $\tilde{\omega} \in \tilde{\Omega}$ and then follows $\tilde{l}$ and when the allocation is 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 $W_{N(\tilde{\omega})}(\tilde{\omega}, \tilde{l}, F^*)$ 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 $F^*$. So we have that

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

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:

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

$$= \left[ \sum_{i : h_i \notin H(\tilde{\omega})} A_i \right] + \sum_{i \in H(\tilde{\omega}), i \neq i'} \alpha_i + p_{i'} A_{i'} + (1 - p_{i'}) \alpha_{i'}$$

$$= \left[ \sum_{i : h_i \notin H(\tilde{\omega})} A_i \right] + \sum_{i \in H(\tilde{\omega})} \alpha_i + p_{i'} (A_{i'} - \alpha_{i'})$$

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

since $h_{i'}$ is the house with maximal option value in $H(\tilde{\omega})$. \textit{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 \( \omega \). An upper bound on \( W_N(\tilde{\omega}, \tilde{l}^*, \tilde{F}^*) \) is calculated as follows:

\[
W_N(\tilde{\omega}, \tilde{l}^*, \tilde{F}^*) = p_{\ell^*} A_{\ell^*} + p_{\ell^*} W_N((\tilde{\omega}, A_{\ell^*}), (\tilde{l}^*, \tilde{F}^*)) + (1 - p_{\ell^*}) W_N((\tilde{\omega}, a_{l^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
= p_{\ell^*} A_{\ell^*} + p_{\ell^*} (p_{\ell^*} A_{\ell^*} + p_{\ell^*} W_N((\tilde{\omega}, A_{\ell^*}, A_{l^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
+ (1 - p_{\ell^*}) W_N((\tilde{\omega}, A_{\ell^*}, a_{l^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
+ (1 - p_{\ell^*}) (p_{\ell^*} A_{\ell^*} + p_{\ell^*} W_N((\tilde{\omega}, a_{l^*}, A_{\ell^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
+ (1 - p_{\ell^*}) W_N((\tilde{\omega}, a_{l^*}, a_{l^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
\leq p_{\ell^*} A_{\ell^*} + p_{\ell^*} W_N((\lambda, A_{\ell^*}), (\tilde{l}^*, \tilde{F}^*))
\]

\[
+ (1 - p_{\ell^*}) W_N((\lambda, a_{l^*}), (\tilde{l}^*, \tilde{F}^*)) = W_N(\lambda, \tilde{l}^*, \tilde{F}^*)
\]

The first equality follows from the learning process \( \tilde{l}^* \) mandating that \( h_{l^*} \) be learned after \( \tilde{\omega} \) and the observation that for any learning history of length \( k \), which includes \( (\tilde{\omega}, A_{\ell^*}) \) as well as \( (\tilde{\omega}, a_{l^*}) \), we have by induction hypothesis that \( \tilde{l}^* \) is the optimal learning process. The second equality follows from \( \tilde{l}^* \) mandating that \( h_{l^*} \) is being learned next since it is the house with lowest index in \( H((\tilde{\omega}, A_{\ell^*})) \) as well as in \( H((\tilde{\omega}, a_{l^*})) \). The third equality is owed to a rearrangement of terms together with the observation that an exchange of the order in which houses \( h_{l^*} \) and \( h_{l^*} \) 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 \( l^* \) and \( \tilde{l}^* \). This implies that

49
\[ W_N(\tilde{\omega}) (\tilde{l}^*(\tilde{\omega}), F^*) = \left[ \sum_{i : h_i \notin H(\tilde{\omega})} A_i \right] + W_N(\tilde{\omega})(\tilde{l}^*, \tilde{F}^*) \]
\[ \geq \left[ \sum_{i : h_i \notin H(\tilde{\omega})} A_i \right] + W_N(\tilde{\omega})(\tilde{l}, \tilde{F}^*) \]
\[ = W_N(\tilde{\omega})(\tilde{l}(\tilde{\omega}), F^*) \]

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 according to which each house \( h_i \) such that the agent who receives it values it at \( \max_{j=1, \ldots, n} \omega_j^i \) 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 that \( i \leq i' \) implies 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 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 Induction we have
\[ p_{i'} A_{i'} + p_{i'} W_N((\tilde{\omega}, a_{i'}, A_{i'}))((\tilde{\omega}, a_{i'}, A_{i'}), \tilde{l}^*, \tilde{F}^*) \]
\[ + (1 - p_{i'}) W_N((\tilde{\omega}, a_{i'}, a_{i'}))((\tilde{\omega}, a_{i'}, a_{i'}), \tilde{l}^*, \tilde{F}^*) < W_N((\tilde{\omega}, a_{i'}))((\tilde{\omega}, a_{i'}), \tilde{l}^*, \tilde{F}^*) \]
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 optimal learning tree if all houses have different option values.

### 11.2 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 that did find it of high value; and no house that was found to be of low value to an agent that 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 3 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 of high value to agents that 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 5** 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 Theorem 1. Let \( (\overline{l}^? , \overline{F}^?) \) be the first best learning process and allocation function in the given problem. To calculate \( \overline{F}^? \) observe that as long as it feasible to follow the prescription of \( \overline{F}^* \) as defined in the proof above it is optimal to do so. posteriori states \( \overline{w} \) that have features: 1. Every agent investigated a house. 2. There exists a house \( h_i \) such that all did find this house to be of low value. 3. No two agents found the house to be of high value. 4. No agent found \( h_{i'} \) of high value. 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 \( \overline{F}^* \) 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 that 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 \( \overline{w} = (a_1, a_1) \), the first best \( (\overline{l}^?, \overline{F}^?) \) 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 \( \tilde{l}^i(\tilde{\omega}) \) 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.

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

Therefore \( \tilde{l}^i(\tilde{\omega}') \) 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 \( \tilde{l}^i(\emptyset) \). If \( \tilde{l}^i(\emptyset) \) prescribes for agent 1 to initiate learning with an investigation of \( h_1 \), then \( \tilde{l}^i(\emptyset) \) does not prescribe for agent 2 to again investigate \( h_1 \) 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 \( \tilde{l}^i(\emptyset) \) prescribes for agent 1 to initiate learning with an investigation of a different house, we also established that \( \tilde{l}^i(\emptyset) \neq l^* \) 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 Theorem 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. 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 that the agent finds $h_1$ 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.

11.3 Hierarchical Exchange Mechanisms

In this subsection I show that Theorem 5 holds for the much larger set of hierarchical exchange mechanism as defined by Papai (2000). 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 multiple 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 that 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.\(^{23}\) A mechanism is said to use **control** if it is not strategically equivalent to a hierarchical exchange mechanism. In other words, control occurs when the outcome of a mechanism cannot also be obtained by assigning property rights and letting agents trade freely.

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

\(^{23}\)This definition of hierarchical exchange mechanisms is sufficient for the purposes of this paper, the interested reader is referred to Papai (2000) for a formally precise definition (which happens to be quite cumbersome).
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.

Note that hierarchical exchange mechanisms leave agents more freedom than mechanisms as defined in this paper. In particular, no firm assumption is made on the order in which agents reveal their decision to point to this or that house. This order can be essential in the present context.\footnote{In case that types are exogenously given the order is irrelevant. This is a result of the strategy proofness of hierarchical exchange mechanisms as shown by Papai (2000).}

The learning decision of some agent might be influenced by whether another agent points to a house owned by him or not. I therefore define **structured hierarchical exchange mechanisms** as hierarchical exchange mechanisms with fixed orders in which agents can decide to point at each other. Probably it would be more in keeping with the market definition that the sequence in which agents point to each other is determined endogenously. This problem is circumvented by proving a negative result through an example of a housing problem in which the first best is implementable, but is not implementable through any structured hierarchical exchange mechanism. This would include in particular a hierarchical exchange mechanism in which the structure of information acquisition is itself owed to an equilibrium argument. We are now ready to address the question, whether for every welfare optimal mechanism there exists a hierarchical exchange mechanism that achieves the same welfare.

**Theorem 6** The second best learning tree and allocation function \((l^o, F^o)\) might not be implementable through a structured hierarchical exchange mechanism.

**Proof** Reconsider the housing problem in Example 3. In this problem the first best \((l^*, F^*)\) is implementable, so \((l^*, F^*) = (l^o, F^o)\). Assume that \((l^*, F^*)\) is implemented via \(s^e\) in some structured hierarchical exchange mechanism \(\Gamma\). W.l.o.g assume that agent 1 starts out owning \(h_1\). I show that \(s^e\) and \(\Gamma\) must be such that agent 1 starts off learning by an investigation of \(h_1\), if \(\Gamma\) is to implement \((l^*, F^*)\) via \(s^e\). To see this, observe that \(h_1\) is the
top ranked house of the a posteriori preferences $\omega_j$ of any agent $j$ that did not investigate $h_1$. Conditional on not investigating $h_1$ agent 1’s best reply is to keep $h_1$ (this entails a utility of $\alpha_1 = 6$ for himself). Defining $p$ as the probability that agent 1 can obtain a different house, agent 1’s utility level of investigating $h_1$ is at least $\frac{3}{4}8 + p\frac{1}{4} \cdot \frac{1}{4} \geq 6$, as he evaluates the different house at least at $\frac{1}{4}$. Next note that $p > 0$, as there is a positive probability that $h_1$ is at the top of any given agent’s ranking.\textsuperscript{25} Therefore it is a strict best reply for agent 1 to investigate $h_1$. Now, since according to $l^*$ there is only one unconditional investigation of $h_1$, it must be that agent 1 starts off the learning tree with his investigation of $h_1$. So assume that agent 1 finds $h_1$ to be of low value. In this case agent 1 will point to $h_2$, the top house according to his a posteriori ranking. If he also owns $h_2$ he will simply leave the market with this house. If not, the situation for the owner of $h_2$ is now as if he owned $h_1$ and $h_2$. By the above arguments it is a best reply for the owner of $h_2$ to investigate $h_1$ and to point to it if he finds it of high value. In this case the owners of $h_1$ and $h_2$ will leave the market with these two houses. It can be concluded that any strategy profile $s^e$ that implements $(l^*, F^*)$ via a structured hierarchical exchange mechanism has the feature that $h_2$ is assigned after a history of investigations according to which $h_1$ was investigated twice and first found to be of low and then of high value. This yields a contradiction as $l^*$ prescribes that $h_2$ should be investigated after such a history. But, of course, no agent can be brought to investigate a house that is already assigned. \hfill $\square$

Observe that I could also have used Example ?? to prove Theorem 6. I showed already in Section 6 that the first best $(l^*, F^*)$ was implementable through a mechanism that has the feature that on the one hand each agent faces a positive probability to end up with house $h_3$ according to the equilibrium strategy profile and on the other hand the initial owner of $h_1$ would always rather appropriate $h_1$ than be assigned $h_3$. However, in that example control and randomization cannot be disentangled: in this example the first best $(l^*, F^*)$ cannot be implemented through a direct choice mechanism. The advantage of using Example 3 in the proof is that this example highlights

\textsuperscript{25}Of course, if the owner of $h_1$ owns another house $p = 1$. 

55
the use of control. Here the mechanism designer can implement the first best through a direct choice mechanism. If, however, the designer was to try to only use randomization as an instrument and to abstain from control, the first best is not achievable. It is consequently 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.

References


