www.sciencemag.org/cgi/content/full/323/5919/1335/DC1

Supporting Online Material for

Promoting Intellectual Discovery: Patents Versus Markets

Debrah Meloso,* Jernej Copic, Peter Bossaerts

*To whom correspondence should be addressed. E-mail: debrah.meloso@unibocconi.it

Published 6 March 2009, *Science* **323**, 1335 (2009)
DOI: 10.1126/science.1158624

This PDF file includes:

Supplementary On-line Material

1 The Knapsack Problem

The knapsack problem (KP) is $\mathcal{NP}-$complete, which means it is $\mathcal{NP}-$hard and every other $\mathcal{NP}-$complete problem can be reduced to the KP (as well as to every other $\mathcal{NP}-$complete problem). The fact that it is $\mathcal{NP}-$hard means that the solution to an instance of the KP can be verified in polynomial time. However, no polynomial-time algorithm is known to *solve* $\mathcal{NP}-$complete problems. The following are the main reasons for choosing the KP as the task faced by participants in our experiment:

1. Even if a participant has seen the KP before, the task is still computationally demanding.

2. Even if an instance can be solved through some simple heuristics, this cannot be guaranteed *a-priori*, without previously having solved the instance.

3. The learning model whereby agents hold a prior over all possible solutions and attempt to solve the problem by gathering 'signals' to Bayesian-update their prior, is not a plausible one (just writing down the likelihood requires that one enumerate all the possible solutions).

4. The fact that the problem involves *items* allows for an intuitive extension to an asset market that does not alter the computational properties of the instance at hand.

1.1 The KP in the experiment

In one experimental session the participants faced eight instances of the KP, under different incentive schemes. The two incentive schemes were the 'Prize' or patent treatment and the 'Market' treatment. In a session there were four prize and four market treatments, in alternating periods, starting with a market treatment. The instances faced by the participants are reported in table 1.

1.2 KP-instance difficulty

There is no straightforward measure of the difficulty of an instance of the KP. For example, even though in the *worst case* an instance of size $n+1$ may take twice as much time as one of size n, this pattern needn't be satisfied by two arbitrary instances of size $n+1$ and n, respectively. In order to compare the instances used in our experiment, we propose two measures of difficulty.

A first intuitive measure is a parameter of *input size*. It is the product of the base-two logarithm of the knapsack capacity times the size of the instance. The base-two logarithm of the knapsack capacity is a proxy for the binary representation of the instance parameters, which in turn represents the amount of storage and information necessary in each step of computation. This is then multiplied by the size (the number of objects) of the instance. In this measure, two instances with equal capacity and size are equally difficult.

Instance & capacity		A	B	C	D	E	F	G	H	I	J	K	L
I	p	500	350	505	505	640	435	465	50	220	170		
	w	750	406	564	595	803	489	641	177	330	252		
$c = 1900$	θ^*	0	0	1	1	0	1	0	0	0	1		
II	p	300	350	400	450	47	20	8	70	5	5		
	w	205	252	352	447	114	50	28	251	19	20		
$c = 1044$	θ^*	1	0	1	1	0	0	0	0	1	1		
III	p	15	14	3	3	10	9	28	28	31	25	24	1
	w	129	144	77	77	66	60	184	184	229	184	219	72
$c = 850$	θ^*	0	0	0	0	1	0	1	1	1	1	0	0
IV	p	37	72	106	32	45	71	23	44	85	62		
	w	50	820	700	46	220	530	107	180	435	360		
$c = 1500$	θ^*	1	0	0	1	1	0	1	1	1	1		
V	p	2	3	4	5	6	9	8	7	6	5	8	9
	w	3	4	6	3	5	13	6	9	2	4	7	7
$c = 14$	θ^*	0	0	0	1	1	0	0	0	1	1	0	0
VI	p	107	35	120	206	88	34	28	110	88	101	74	53
	w	599	196	670	1204	502	202	145	600	453	601	404	299
$c = 3800$	θ^*	1	1	0	0	1	0	1	1	1	1	1	1
VII	p	201	84	113	303	227	251	129	147	86	127	144	167
	w	192	80	106	288	212	240	121	140	82	120	137	160
$c = 1300$	θ^*	1	0	1	1	1	0	0	1	1	1	0	1
VIII	p	31	141	46	30	74	105	119	160	59	71		
	w	21	97	32	21	52	75	86	116	43	54		
$c = 265$	θ^*	0	1	0	0	1	0	0	1	0	0		

(Columns A–L are grouped under the header "Objects".)

Table 1: Instances of the knapsack problem used in experimental sessions. Objects had common names, not letters. The letters used in the table stand for: A=Anderson, B=Brown, C=Cole, D=Darwin, E=Evans, F=Foster, G=Green, H=Hamilton, I=Ives, J=Jensen, K=Keaton, L=Lee.

The second proposed measure relates to the question of heuristic solvability of an instance. A simple approximation algorithm for the knapsack problem may solve an instance exactly. For a class of approximation algorithms described below, we consider an instance that can be exactly solved with an approximation algorithm in this class to be easier than another instance that cannot be exactly solved with it.

The simplest approximation algorithm for the knapsack problem is the *greedy* procedure, which consists of filling the knapsack in efficiency order - i.e. starting with the objects that have a higher $\frac{v_i}{w_i}$ ratio - until the weight limit is reached. This heuristic is part of a family of approximation algorithms known as the Sahni approximation scheme [2]. The Sahni scheme is parameterized by a number k, referring to the specific algorithm, which we call a *Sahni algorithm of size* k.

A Sahni algorithm of size k looks at all subsets of the set of all objects that are considered to enter the knapsack, of cardinality k or less. For each subset, it computes the residual weight in the knapsack after subtracting the weight of the subset, and fills this residual with the remaining objects using the greedy procedure (if no set of k objects fits in the knapsack, then the exact optimal solution is found by the algorithm). The value of all knapsacks constructed in this way is compared, and the one with highest value becomes the approximate solution given by the algorithm. Clearly, the greedy algorithm is a Sahni algorithm of size 0. The Sahni algorithm of size 1 uses the greedy algorithm for every subset of size $n-1$ (there are n such subsets) to fill the capacity that remains after isolating one object. Though the complexity of the greedy algorithm run by a Sahni-1 algorithm is smaller than the greedy algorithm run for the original instance (Sahni-0), the Sahni-1 algorithm must run the greedy algorithm n times. The complexity added is thus of order n. This is the case for every increase from k to $k+1$. Although a Sahni algorithm of size k is not as straightforward as the greedy algorithm, it is still a very simple heuristic.

Definition 1 *We say that an instance has Sahni-difficulty level k if it can be exactly solved with a Sahni algorithm of size k, but not with a Sahni algorithm of size $k-1$. The higher the k associated to an instance, the harder the instance.*

While specific instances can be solved using simple heuristics, it is impossible to determine a-priori what simple heuristic to use. Only after the instance has been solved can one know what Sahni algorithm to use. Thus, there is no reason to believe that Sahni-k will be a good predictor of whether a person can solve an instance or not.

The intuitions behind the two proposed measures are simple. Suppose a person uses an obvious *exact* algorithm to solve the KP: he computes the value and weight of every possible solution, then he checks for feasibility, and chooses the highest value among the feasible solutions. In this case, he should find an instance with a lower *input size* to be easier than one with a higher input size. The time he requires to compute the former should be smaller than the time he requires to compute the latter. On the other hand, suppose a person uses a simple approximation algorithm, taken from the Sahni family, to solve the KP. Even though an instance with a higher input size is still harder to deal with, the effect of the input size is relatively small. Instead, this person will face the problem that he will not find the correct solution to instances that have a difficulty level that is larger than the k of the algorithm he uses. Hence, even though his computation time is

	Input Size Proxy ($n \log_2 c$)					
		40 − 50	80 − 90	100 − 120	120 − 130	140 − 150
Sahni-difficulty level	0			IV		
	1	V		I		
	2		VIII	III		
	3			II		VI
	6				VII	

Table 2: Difficulty of Knapsack Problem instances used in experiments. The difficulty is measured by the proxy of *input size* (columns) and the Sahni difficulty level.

almost invariant, he will only correctly solve instances with a small enough Sahni-k difficulty level.

Table 2 describes the difficulty of each instance used in the experiment, as measured by both above-mentioned measures. The first intuitive measure ($\log_2 c$) proved to be of little relevance to our experimental results. Specifically, it did not correlate with the number of people who solve an instance within a treatment. The Sahni-k measure proved insightful, as described in the main article.

2 Experimental Design: Additional Details

2.1 Trading mechanism and software

The trading mechanism used in the market treatment was an open-book double auction. This mechanism has the following properties:

- The markets for each of the tradable securities is open during a fixed amount of time.
- While a market is open, participants can submit limit orders to buy (bids) a given number of units at a given maximum price, or limit orders to sell (asks) a given number of units at a given minimum price. Limit orders do not match an outstanding order and, therefore, become part of a publicly known list of orders (the order book).
- While a market is open, participants may execute trades by sending a market order to buy that matches (has a price larger than or equal to) an outstanding (limit) order to sell, or sending a market order to sell that matches (has a price smaller than or equal to) an outstanding order to buy.
- The trading price is given by the limit order that is executed.
- All orders and transactions are anonymous.
- When markets close, dividends are realized according to a pre-specified rule. In our case, the rule relates to the solution of the instance of the KP that is being considered. (In

asset-pricing experiments, the rule is a specific random variable. In commodity-trading experiments, the rule is a specific private value.)

Additionally, the specific market that we implemented had the property that each participant could both buy and sell (they did not have assigned buyer or seller roles), participants could trade several units of each security and, in particular, they could re-trade (resell units previously acquired). However, participants could not sell short.

Double-auction markets have been extensively studied experimentally, and have proven to allow for robust convergence of prices to theoretical competitive equilibrium prices in commodity markets (see [3]), and asset markets (see [1]).

The software used to implement the double auction described above, was eTradeLab.[1] This software displayed the best bid and ask outstanding in all markets, and could display the entire book for a specific market if requested by the user (through a click). To submit a limit order, participants had to choose a market (an asset they wish to bid for), choose a type of order - order to buy or order to sell - and type in a number of units and a limit price for the transaction. To submit a market order, participants could proceed in the same way as to submit a limit order, making sure the submitted price matched an existing limit order, or they could click directly on the marker for best bid or best ask in one of the asset markets. This would immediately generate a transaction. During an experimental session, all transactions were made and recorded in an experimental currency called *Franc*, with an exchange rate of $100 Franc/\$$.

The more recently developed *jMarkets* is publicly available and can be used for replication of our experiment. The experimental sessions run using *jMarkets* (robustness check, see section 4) used US Dollars as the experimental currency.

2.2 Implementation of Market Treatment

In the market treatment, subjects were all endowed with an equal number of units of each security. That is, all subjects started out with 5 units of each traded security. Each security's dividend was attached to an item in the KP: if the item was part of the optimal solution, then the security paid a dividend of $1, it paid $0 otherwise. Additionally, each subject was endowed with $4 to start trading in the markets.

Clearly, the above endowment combined with the average number of items in the optimal solution in our instances of the KP, would lead to a large expected payoff for each subject. It was, however, necessary to give a large endowment, since without it (and without the possibility to sell short) there would not be enough liquidity to start trading in the markets. In order to ensure that the total payoff across all subjects in the market treatment equaled the total payoff in the prize treatment (on average), we used a *loan repayment*. In type *a* sessions, where instances I, V, VII, and VIII were solved in the market treatment, the loan repayment was $23.75. In type *b* sessions, where instances II, III, IV, and VI were solved in the market treatment, the loan repayment was $32.50. This meant that subjects knew in advance that at the end of every market period, their

[1] This software, a prototype of jMarkets (jmarkets.ssel.caltech.edu), was developed by Tihomir Asparouhov. It is based on the Marketscape software developed in Charles Plott's laboratory at Caltech.

earnings would equal the earnings from trade and dividends, minus the fixed loan repayment (the exact amount of the loan repayment was also known in advance).

The following simple calculations show that the combination of initial endowment and loan repayment led to an average total payoff of approximately \$66 in the market treatment: Each subject held 5 units of each asset. Since the number of subjects in our sessions was between 15 and 17, this translated to a total across students of between 75 and 85 units of each asset in a single market-treatment period. In type *a* sessions, the optimal knapsacks for each of the instances solved in the market treatment contained 4, 4, 8, and 3 items, respectively. The assets corresponding to these items paid a dividend of \$1 per unit. Thus, the total dividend that was distributed between the subjects was $4 \times 75 = \$300$ (\$340 in the case of 17 subjects) for instances I and V, $8 \times 75 = \$600$ (\$680 in the case of 17 subjects) for instance VII, and $3 \times 75 = \$225$ (\$255 in the case of 17 subjects) for instance VIII. To each of these dividends we must add the initial cash holdings of $\$4 \times 15 = \$60$ (or $\$4 \times 17 = \$68$), leading to a total of between \$360 and \$408 for instances I and V, between \$660 and \$748 for instance VII, and between \$285 and \$323 for instance VIII. Therefore, the average over all market-treatment periods of the total payment that was divided among subjects *before loan repayment* lied between \$416.25 and \$471.75. The total loan repayment per period lied between $\$23.75 \times 15 = \$356.25$ and $\$23.75 \times 17 = \$403.75$. Hence, on average (average taken across market-treatment periods), subjects had a total fund of between $\$416.25 - \$356.25 = \$60$ and $\$471.75 - \$403.75 = 68$, to divide among them in a particular market-treatment period. This interval includes \$66, which is the amount of money given as the prize in a prize-treatment period. Analogous calculations lead to the same conclusion for type *b* sessions.

2.3 Implementation of Prize Treatment

In the prize treatment, participants were given the instance statement and were allowed to use only pen and paper to attempt a solution in a fixed window of time. If a participant desired to submit a solution, he or she had to raise his or her hand to call the attention of the experimenter. Each participant was allowed to submit only one solution, since the experimenter's review of his or her solutions could give different participants privileged information about the correct solution. If a participant submitted a correct solution, this immediately became public knowledge.

If two (or more) participants raised their hands at moments that were undistinguishable to the experimenter, and both provided correct answers, a tie was called. In such a case, the prize of \$66 was shared between the tied participants.

2.4 Sessions and subjects

In the main article we report on four experimental sessions run in the end of 2004. All sessions were run at Caltech, using students as subjects. Three of these sessions had 15 participants, while the remaining session had 17. The alternative treatment reported below was run in four sessions in the end of 2006 and beginning of 2007. Two sessions had 14 participants each, one session had 16, and another had 18 participants.

		Session			
		040809a	040929b	041202a	041215b
Average per person per period	Asks	6.97 (5.1)	5.38 (3.6)	7.68 (4.4)	5.3 (4)
	Bids	6.28 (3.7)	5.15 (2.9)	6.07 (4.2)	5.03 (3.7)
	Trades	5.43 (3.8)	4.19 (3.5)	5.63 (3.2)	4.07 (2.7)
	Volume	11.07 (8.5)	8.79 (6.9)	12.6 (7.8)	12.88 (9.2)
Average per security per period	Asks	8.71 (4.1)	7.62 (3.6)	9.6 (4.8)	6.62 (3.9)
	Bids	7.85 (5.3)	7.29 (3.9)	6.07 (4.2)	6.29 (3.7)
	Trades	6.79 (3.8)	5.94 (4.0)	7.04 (3.2)	5.08 (3.8)
	Volume	13.83 (9.9)	12.46 (9.0)	15.75 (9.8)	16.1 (11.9)
Totals	Asks	418	366	461	318
	Bids	377	350	364	302
	Trades	326	285	338	244
	Volume	664	598	756	773

Table 3: Summaries of trade and bidding activity.

3 Results: Additional Information

Here we report graphically the results for individual experimental sessions. We group specific periods of several sessions according to the instance of the KP that is given to participants in that period.

3.1 Trade volume

Tables 3 and 4 summarize information about trading and bidding volume.

3.2 Prices and Holdings

Figures 1 to 8 contain three panels each. The first two panels show the evolution of asset prices in experimental time for each experimental session where the instance at hand was solved under the market treatment. Assets are divided in 'IN' and 'OUT'. 'IN' assets are those related to items that pertain to the set that solves the instance. 'OUT' assets are those related to items that are

		Session			
		040809a	040929b	041202a	041215b
Trade volume	IN	15.4	10.7	17.3	12.5
	OUT	14.9	17.7	17.12	24.9
Asks	IN	7.9	6.0	9.5	5.3
	OUT	10.7	11.7	11.2	10.1
Bids	IN	10.9	8.0	9.8	6.3
	OUT	6.8	7.9	7.1	7.7

Table 4: Volume of trade and number of bids and asks, for IN and OUT securities.

left out of the set that solves the instance. The third panel shows the empirical pdf of the prices of 'IN' and the prices of 'OUT' securities, for the instance at hand. Not all securities were traded at every moment in time, and some securities were traded only once (they are displayed as a dot, not a line, in the figure).

Prices of 'IN' and 'OUT' securities almost never equaled 1 or 0 respectively, because one could never be *sure* whether a single item was in or out of the optimal solution without knowing the optimal solution. To be sure that one knew the optimal solution one had to solve the instance *exactly*, meaning that all the possible solutions needed to be listed and compared. Also, one should refrain from interpreting the prices as conditional probabilities of whether the item is in or out of the optimal solution. Computation of this probability requires that one know the likelihoods, which in itself pre-supposes a list of all possible solutions and their values. If a subject were able to produce such a list and compare values in the limited time s/he had, then s/he would also be able to compute the value of each possible knapsack and hence, produce the exact optimal solution. Bayesian updating of a prior in order to produce probabilities of items' belonging to the optimal solution, is simply not a viable model of learning in the KP.

Figure 9 shows the histogram of final holdings of the participants for every instance of the KP. For every instance, the histogram is plotted for each asset (one can think there is one histogram "per column"). Even though a holding of 5 units of an asset is always the modal number of units, it is easy to see that holdings take on many different values. This also means that there was in general a substantial amount of trade, because all subjects started out with the same initial allocations (5 units).

4 Alternative experimental setup

In addition to the experiments reported in the main body of the article, four experimental sessions were run that differed in several ways from the original experimental setup (instructions for this alternative setup can be found at http://clef.caltech.edu/exp/knapsack/). The market treatment differed in the following aspects:

- The time markets were open was reduced from 15 to 10 minutes. The reason for this

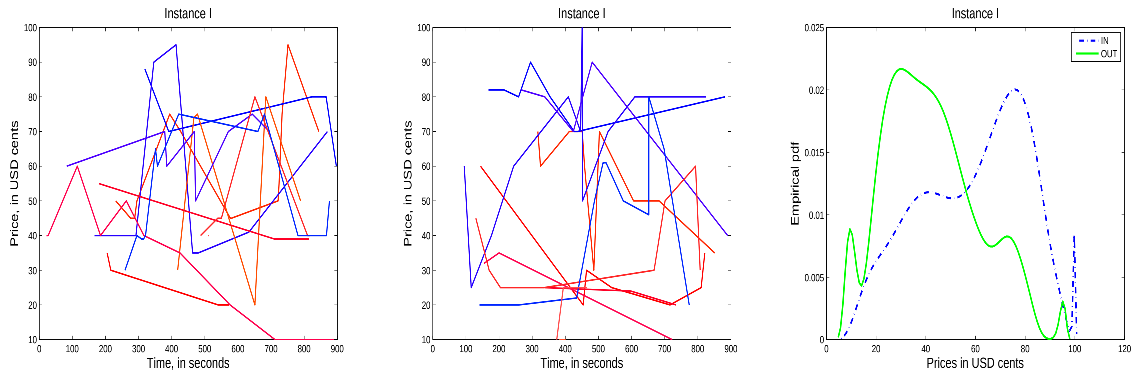


Figure 1: Instance I. The red lines are prices of 'OUT' securities, the blue lines of 'IN' securities.

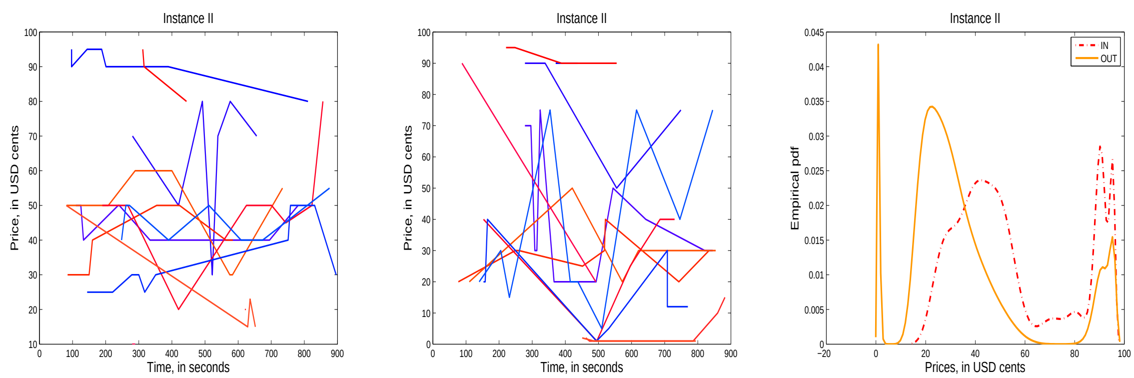


Figure 2: Instance II. The red lines are prices of 'OUT' securities, the blue lines of 'IN' securities.

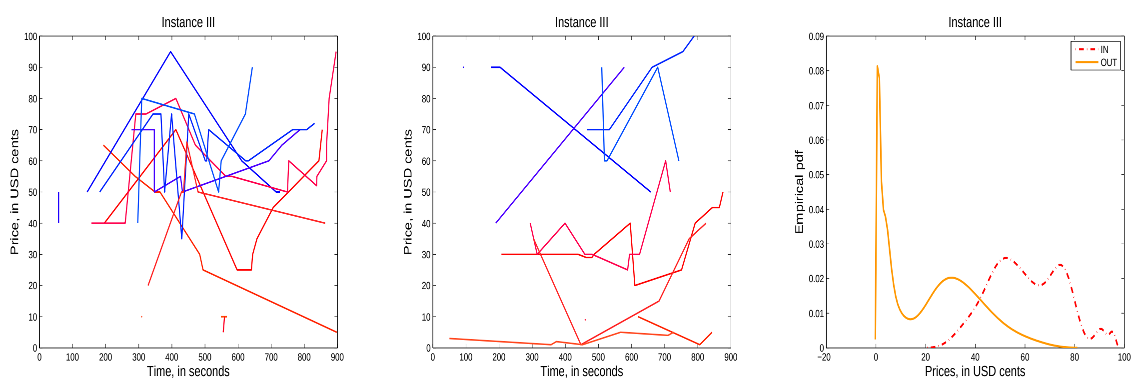


Figure 3: Instance III. The red lines are prices of 'OUT' securities, the blue lines of 'IN' securities.

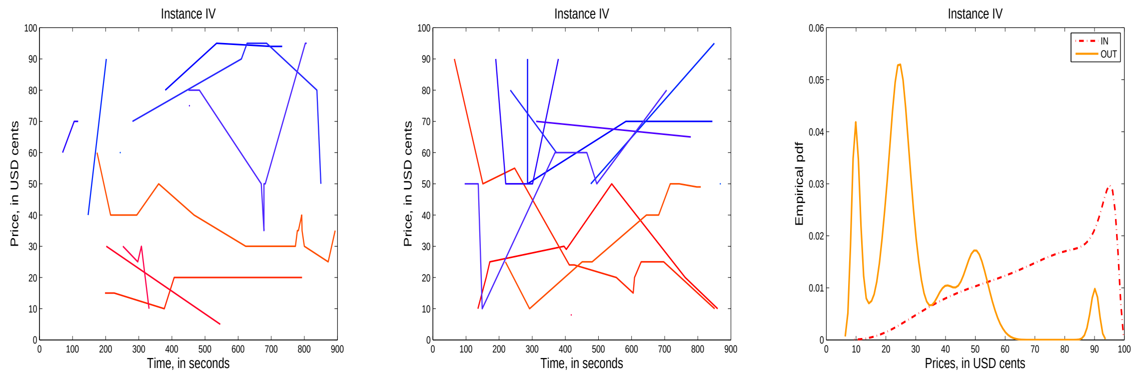


Figure 4: Instance IV. The red lines are prices of ‘OUT’ securities, the blue lines of ‘IN’ securities.

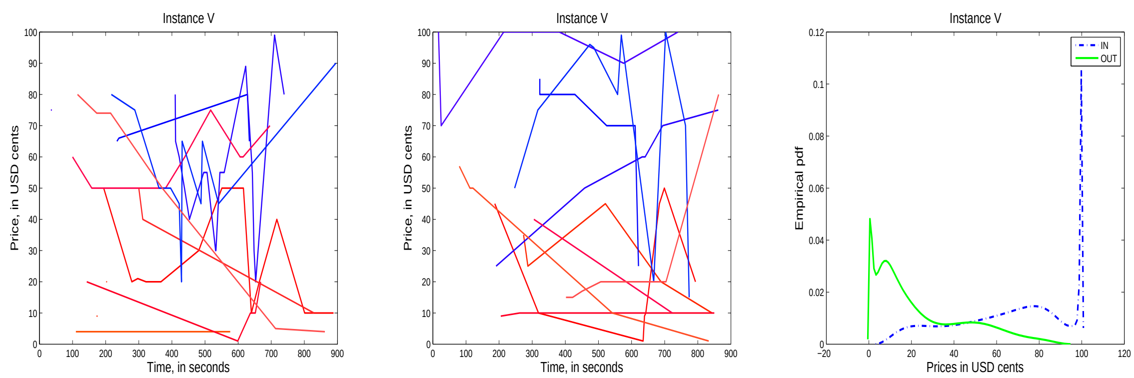


Figure 5: Instance V. The red lines are prices of ‘OUT’ securities, the blue lines of ‘IN’ securities.

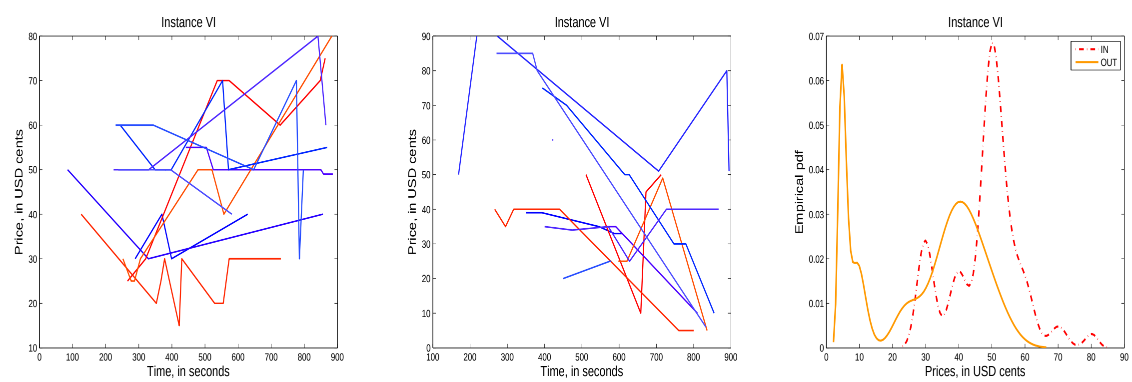


Figure 6: Instance VI. The red lines are prices of ‘OUT’ securities, the blue lines of ‘IN’ securities.

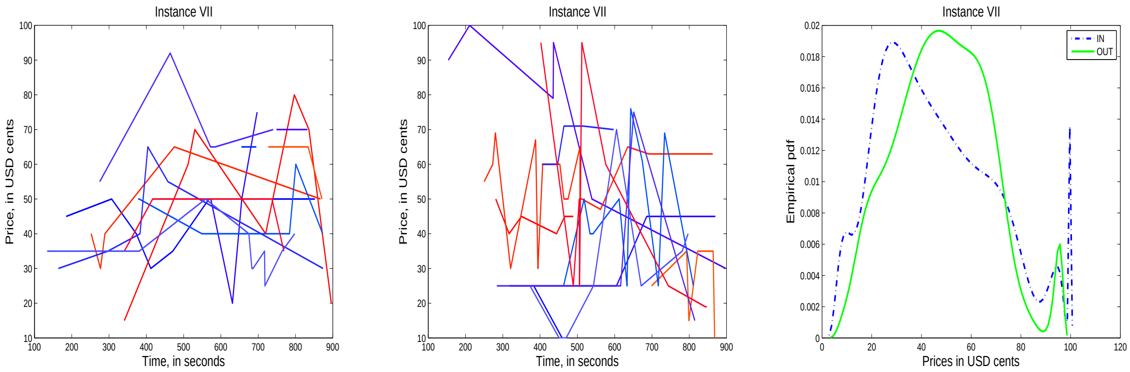


Figure 7: Instance VII. The red lines are prices of 'OUT' securities, the blue lines of 'IN' securities.

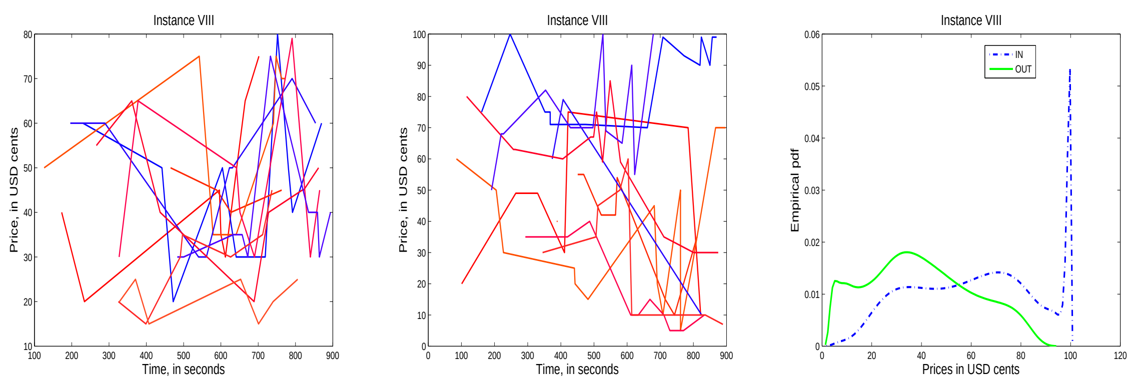
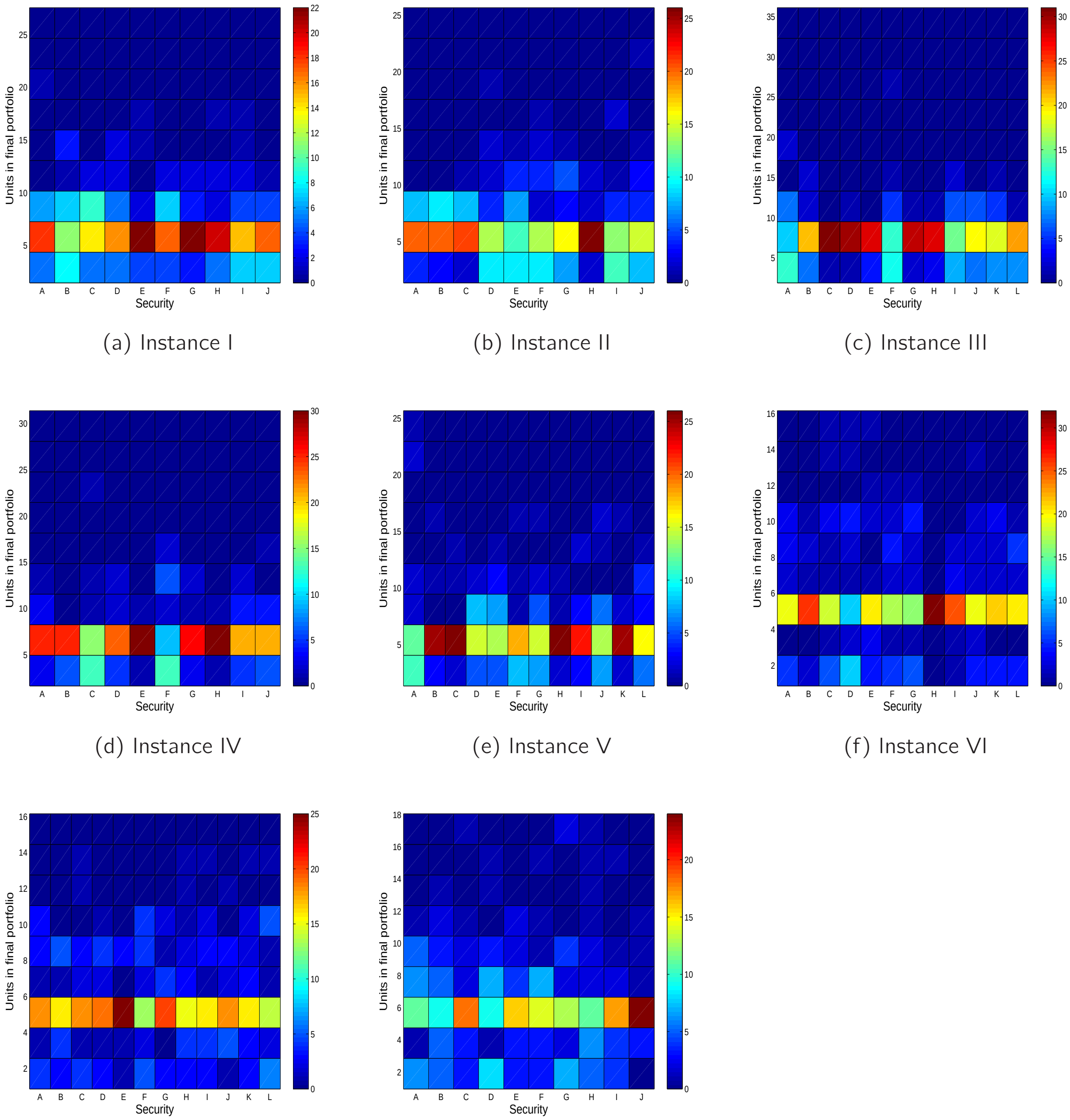


Figure 8: Instance VIII. The red lines are prices of 'OUT' securities, the blue lines of 'IN' securities.



(a) Instance I (b) Instance II (c) Instance III

(d) Instance IV (e) Instance V (f) Instance VI

(g) Instance VII (h) Instance VIII

Figure 9: Histogram of final security holdings for each instance of the KP (aggregate over experimental sessions). The colors indicate the frequencies of given number of units for each security.

change was to match the duration of the markets and the prize periods, while keeping the total duration of an experimental session manageable.

- The market software used was changed to jMarkets. This was done mainly because the quality of jMarkets was superior to that of the previously used software (eTradeLab); it is more reliable, faster, and especially easier for subjects (all order submissions are "point and click", so no mistakes occur because of typing of numbers). It is important to point out that the difference between the two softwares are only of "form", and not of substance. Both softwares implement exactly the same type of financial markets.

On the other hand, the prize treatment was changed in more significant ways. The main changes were the following:

- The maximum duration of the prize-treatment period was raised from 7 to 10 minutes. In this way, the duration of the market and prize periods was equalized.
- Instead of raising their hand, participants had to submit their answers to the experimenter using a *chat* program called skype. The program immediately time-tagged the submission, and fixed it in the experimenter's computer screen.
- If a correct solution was received by the experimenter over skype at a time before the end of the period, this was NOT announced. The time tag of the winning submission (first correct answer submission) was recorded, but other participants were unaware that there was already a winner.

In terms of setup and results, the third bullet above is the most relevant. We must remark, however, that this pushes the *prize* treatment away from a standard patent system: once a discovery is patented, it is publicly known, and it becomes unprofitable for others to continue pursuit of the discovery.

4.1 Results of alternative setup

The main change in results from the original to the alternative setup is in the number of voluntarily-handed in solutions that are correct under the prize treatment. It is no longer lower than the number of (voluntarily-handed in) solutions under the market treatment. Instead, the numbers described in table 5 emerge. Unlike in the baseline setup, there is not a clear winner: the markets and prize treatments perform equally well.

However, the prices in the market treatment of the alternative setup show an even clearer separation of 'IN' and 'OUT' securities. This is probably because the software is easier to use; specifically, typing mistakes cannot occur, by design. Here we only show the probability density functions (fig. 10).

The correlation between instance difficulty and the number of participants that find the solution in each treatment is still patent (fig. 11). It is, however, difficult to make a comparison between this figure and the analogous figure (displayed in the main article) for the original experiment. Firstly, the prize treatment differs from the original experiment in an important aspect. Unlike

		Instance							
		I	II	III	IV	V	VI	VII	VIII
Fraction Correct (%)	Market	32.14	14.71	11.76	26.47	32.14	5.88	3.57	21.43
	Prize	17.65	28.57	46.43	57.14	38.24	10.71	0.00	17.65

Table 5: Percentage of total number of participants that marked the correct answer to each instance on their answer sheets – alternative setup.

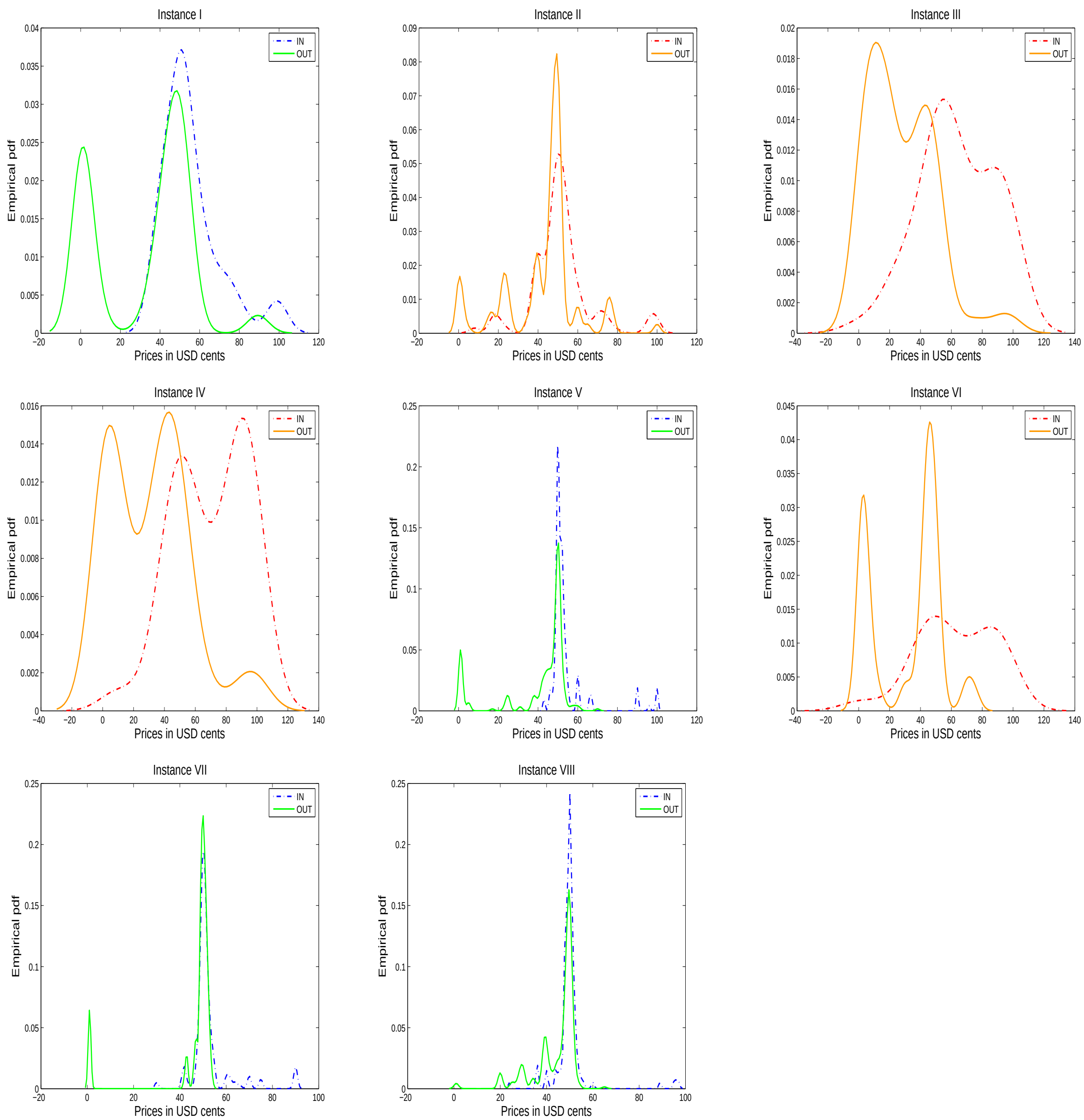


Figure 10: Empirical pdf of prices of 'IN' and 'OUT' securities for all instances of the KP. Results from the alternative experimental setup.

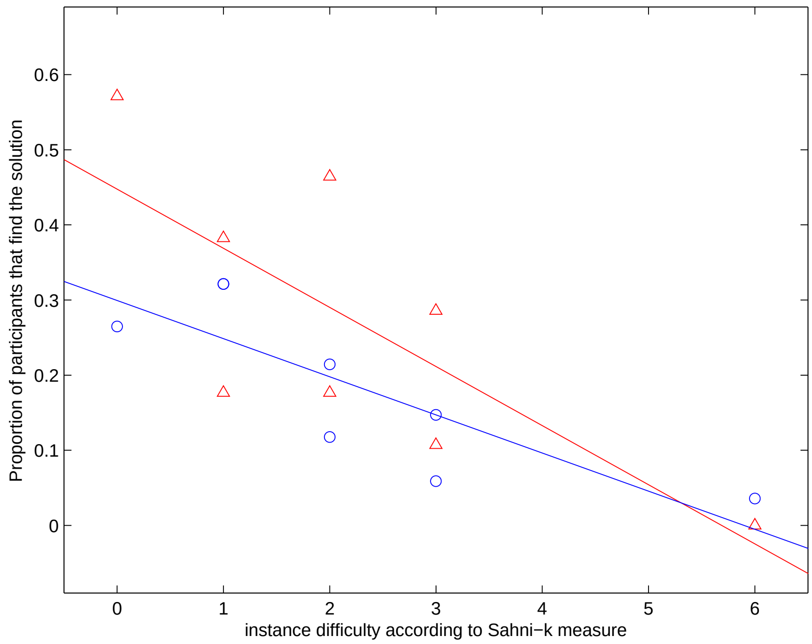


Figure 11: Fraction of all participants that found the correct solution to an instance, as a function of Sahni difficulty of the instance. The lines are the OLS fits of the data points. Red triangles and the red line correspond to the prize treatment, while the blue circles and blue line correspond to the market treatment.

in the main experiment (and a true patent system), we did not announce whether the optimal solution was found until the end of the period. Yet only the first subject to find the solution was paid; this feature provided explicit monetary incentives for subjects to continue to work on solving the problem even after someone had found the solution.

Secondly, in the alternative setup we had a large number of non-respondents, which obscured the interpretation of response data. Specifically, the fact that in the alternative setup the markets system did not get any time advantage, substantially decreased the number of subjects that handed in their suggested solution in the markets treatment: more than 1 out of 4 subjects now became non-responsive and never handed in anything. This made it difficult to objectively interpret the results of the second set of experiments: the performance of the markets system was likely to be underestimated, but we did not know by how much. Now, when interpreting non-responsiveness as an indication that the subject did not know the correct answer, the performance of the markets system was indeed reduced – but only for easy knapsack problems. This is clearly visible in figure 11.

References

[1] Peter Bossaerts and Charles Plott. Basic principles of asset pricing theory: Evidence from large-scale experimental financial markets. *Review of Finance*, 8(2):135–169, 2004.

[2] S. Sahni. Approximate algorithms for the 0-1 knapsack problem. *Journal of the ACM*, 22:115–124, 1975.

[3] Vernon Smith. An experimental study of competitive market behavior. *The Journal of Political Economy*, 70(2):111–137, 1962.