ELSEVIER

Journal of Economic Behavior & Organization
Vol. 50 (2003) 249–262

JOURNAL OF
Economic Behavior
& Organization

www.elsevier.com/locate/econbase

An ordering experiment

A. Norman*, M. Ahmed, J. Chou, K. Fortson, C. Kurz, H. Lee, L. Linden, K. Meythaler, R. Rando, K. Sheppard, N. Tantzen, I. White, M. Ziegler

Department of Economics, The University of Texas at Austin, Austin, TX 78712-1173, USA

Received 12 December 2000; accepted 20 April 2001

Abstract

Binary comparison operators form the basis of consumer set theory. If humans could only perform binary comparisons, the most efficient procedure a human might employ to make a complete preference ordering of n items would be a $n \log_2 n$ algorithm. But, if humans are capable of assigning each item an ordinal utility value, they are capable of implementing a more efficient linear algorithm. In this paper, we consider six incentive systems for ordering three different sets of objects: pens, notebooks, and Hot Wheels. All experimental evidence indicates that humans are capable of implementing a linear algorithm, for small sets.

JEL classification: C9 Design of Experiments; C91 Laboratory, Individual Behavior; D11 Consumer Economics: Theory; C61 Optimization Techniques; Programming Models; C63 Computational Techniques; C69 Other-Complexity Theory

Keywords: Linear algorithm; Binary comparison; Ordering

1. Introduction

We experimentally examine the algorithms that individuals use to order sets of objects. Our focus is not the static properties of the ordered objects, but rather the properties of the algorithms that humans use to make the ordering. By comparing human algorithms with known computer ordering algorithms, we can determine their properties. We use complexity theory to measure cost and to formulate hypotheses to test among alternative human algorithms.

If humans were limited to a binary ranking operator in determining a complete preference ordering for a set of n items, humans could employ an algorithm no more efficient than

* Corresponding author.
E-mail address: norman@eco.utexas.edu (A. Norman).

0167-2681/03/$ – see front matter
PII: S0167-2681(02)00050-1

$n \log_2 n$ binary ranking operations in both the worst and expected cases.We discuss several of these sorting algorithms and provide a brief survey of computational complexity in Section 2. However, if consumers were able to assign an ordinal utility value to each item in the set, they could employ a conceptually simple linear bucket sort algorithm.

The experimental design for ordering sets of pens, notebooks and Hot Wheels and the problem of incentives are described in Section 3 and the observations of the subjects' behavior are presented in Section 4. The observations and regression results presented in Section 5 show that the subjects are using the linear modified bucket sort to order the selected objects. In Section 6, our ordering data also indicates that humans are not consistent in their orderings of pens. Finally, we conclude in Section 7.

2. Computer ordering

In this section, we examine ordering algorithms, commonly referred to as sorting algorithms, Aho et al. (1974), Knuth (1973), Mehlhorn (1994). In this paper we asymptotically compare algorithms on the time to sort as the number of items, n, increases. Roughly speaking, worst case (expected value) analysis is finding the algorithm which has the best worst case (expected value) performance for all possible (specified distribution of) initial positions of the objects to be sorted.

To define the asymptotically computational complexity, let $Y = Y(n)$ be a nonnegative function which we wish to compare with the cost function, $C = C_\varphi(n)$. Frequently Y is n, n^2 etc. Consider the following definitions:

D1. C is $O(Y)$ if there exist $i, j > 0$ such that $C(n) \leq jY(n)$ for all $n > i$.
D2. C is $\Omega(Y)$ if there exist $i, j > 0$ such that $C(n) \geq jY(n)$ for all $n > i$.
D3. A problem has computational complexity Y if there exists an efficient algorithm φ_i such thatC_{φ_i} is $O(Y)$ and for all algorithms φ_j C_{φ_j} is $\Omega(Y)$.

In order to discuss sorting algorithms consider a set $B_n = b_1, b_2, \ldots, b_n$ of close substitutes. First, we characterize algorithms based on an information operation to determine the attributes of each good and a binary comparison operator. We define the information operator $D(b_i) = a_i$. Because subjects do not determine the position of each molecule on the surface of the test objects, the information operator returns a truly minuscule fraction of the potentially observable attributes. Indeed, we assume like Myers and Alpert (1968) the subjects employ an information operator that determines only those attributes necessary to make comparisons. For a discussion of the affect of varying amounts of information on decisions, see Payne et al. (1993). Minor variations in the information operator among subjects are considered part of the error term in the regressions.

Subjects use the information operator in order to execute a ranking operator, $R(a_i, a_j) \Rightarrow b_i \succeq b_j$ or $b_j \succeq b_i$ that determines the preferences between two items, b_i and b_j. We do not assume transitivity in this paper because of the increasing empirical evidence of intransitive preferences starting with Tversky (1969). Intransitivity will not affect the complexity of the algorithms, just the extent to which the orderings have desirable properties.

We assume each type operator $D(b_i)$ and $R(a_i, a_j)$ incurs a cost of c_D and c_R respectively. These costs are considered to be constant both over time and over the entire set of items. We also assume that each type operator $D(b_i)$ and $R(a_i, a_j)$ is performed in a fixed time of t_D and t_R, respectively. Thus, we are assuming an equivalence between cost and time complexity, a standard assumption in complexity analysis.

In discussing sorting algorithms we assume that humans will employ the most efficient algorithm they are capable of implementing. As computer scientists have spent several decades developing sorting algorithms, it is unlikely that humans intuitively use an algorithm not mentioned in this section.

Thus, we might expect humans to use a variation of a bubble sort to order a set of pens. A possible algorithm to order the pens in a complete preference ordering from left to right is as follows:

(1) write with the first pen, $D(P_1) = a_1$, and place on the table;
(2) write with the second pen, $D(P_2) = a_2$. If $R(a_1, a_2) \Rightarrow P_2 \succeq P_1$, place P_2 to the left of P_1, else place P_2 to the right of P_1;
(3) now consider the placement of the mth pen after the $m-1$ pens have been ordered. Starting with the pen furthest to the right, compare P_m with each pen moving to the left until the following condition is satisfied: $P_m \succeq P_i$ and $P_{i+1} \succeq P_m$. Place P_m between P_i and P_{i+1}. If $P_m \succeq P_{m-1}$ place P_m to the left of P_{m-1};
(4) repeat (3) until all pens are ordered.

Because this algorithm requires $1 + 2 + 3 + \cdots + n - 1$ ($=n(n-1)/2$) binary comparisons in the worst case and half that number in the expected case, it is $O(n^2)$ both in the worst and expected cases. The absolute cost, but not the asymptotic cost of this algorithm can be reduced by starting the comparison of each new pen with the existing ordering in the middle and proceeding up or down depending on the pairwise comparisons. If the sorter started in the middle, on average he would only have to compare the new pen with 1/4 of the pens in the existing ordering.

Efficient algorithms exist for ordering which are only $O(n \log_2 n)$ in the worst and expected value cases, Mehlhorn (1994). Let us consider one of the algorithms efficient in the worst case (and expected case), Mergesort. A human might employ this algorithm to order 16 pens in the following manner:

(1) write with all sixteen pens;
(2) lay the pens in a row;
(3) merge the first with the second, the third with the fourth, ..., the 15th with the 16th. Merging the first with the second requires comparing the first with the second and ordering them in a column with the preferred item first. This results in eight columns of two members each;
(4) merge the first column with the second, the third with the fourth, ..., the seventh with the eighth. This step results in four columns of four members each;
(5) in a similar fashion, merge the four columns merged into two columns of eight members each;
(6) merge the two columns into one ordered column.

Another efficient algorithm in the worst case is the Heapsort algorithm and a third algorithm, efficient in the expected value case, is the Quicksort algorithm. Because each of these algorithms take considerable time to understand, they are not likely to be used intuitively. Nevertheless, if the subjects were using one of these algorithms it would be observable from the way they organized the objects and from their hand motions.

But, if humans can assign ordinal utility values to items in a set, then humans could use a more efficient algorithm. Consider the bucket ordering algorithm used to sort it n integers all contained in the interval (k, l), where k and l are integers. This sorting process uses the following algorithm for $(l - k - 1)$ integers on the interval:

(1) create $(l - k - 1)$ buckets;
(2) pick an integer and determine its value;
(3) place it in the bucket with the corresponding value;
(4) repeat (2) and (3) until all numbers are assigned to buckets.

The integers are now sorted. Thus, a human wishing to efficiently sort a set of integers could use a bucket sort. This idea, however, is not limited to sorting integers. Consider a human ordering n items in a set B. If a human can use an information operator $G(b_i)$ to assign an ordinal utility value to each item, then the human could use a placement operator $P(b_i)$ to place each b_i in the correct bucket. The number of G and P operations is $O(n)$ for this bucket sort, thus the computational complexity is n in time and cost assuming each operation has fixed unit cost and time. Defects in assigning ordinal utility values would affect the consistency of the outcome, but not the efficiency of the algorithm. A bucket sort is the only known linear algorithm for sorting.

In determining what algorithm subjects use we shall:

(1) observe subjects' object organization and the hand motions;
(2) analyze the regression results;
(3) ask the subjects questions.

3. Experimental ordering design

3.1. Groups and subgroups

To determine how humans actually establish a complete preference ordering, we selected 320 University of Texas at Austin undergraduates in ten groups of 32 divided into subgroups of eight subjects. Each subject was given a practice set of three objects to sort to ensure that they were following the instructions. Then each subject ordered three sets of objects as shown below:

Groups, subgroups and the size of the three sets to order

Group	Subgroup			
	1	2	3	4
1–6	(3, 6, 17)	(3, 7, 16)	(3, 8, 15)	(3, 9, 14)
7–10	(3, 5, 9)	(3, 6, 10)	(4, 7, 11)	(4, 8, 12)

Thus, each of the eight members of subgroup 3, group 9 orders sets of 4, 7, and then 11 objects. The numbers were chosen to provide data over many numbers for the regression analysis.

3.2. Objects

We choose pens, notebooks, and Hot Wheels for objects that student experimenters thought student subjects would be familiar. We made a concerted effort to ensure that the objects were distinctive. We included pens with various types of points, ink, colors, size, style and cost. Similarly, we selected notebooks on which the students might take their class notes, including notebooks of different sizes from 3×5 to 8.5×11, color of paper, binding, writing pads such as legal pads and we even included a clipboard and three-ring binder. From a very large selection of Hot Wheels, we selected sports cars, station wagons, pickups, a garbage truck, and even a tank. In selecting sets of objects to order we made an effort to make the sets representative of the various types of objects in the overall set.

Subjects performed strong and weak orderings on the listed objects:

Objects for experiment

Group	1	2	3	4	5	6	7	8	9	10
Ord	S	W	S	S	S	S	S	S	S	S
Obj	P	P	P	P	P	P	P	NB	HW	HW

where Ord, S, and W are ordering, strong, and weak, respectively. Obj, P, NB, and HW are object, pens, notebooks, and Hot Wheels, respectively.

3.3. Evaluation

There are two processes in this experiment i.e. the evaluation of objects and the ordering or placement of objects. Since our focus is ordering, we want the subject to evaluate each object in an experiment in approximately the same time. In an earlier version of the experiment, subjects start writing their signatures initially with each pen and then gradually reduce the amount they wrote to a short wiggle. In order to ensure that subjects evaluated each object in a uniform time, subjects were asked to write 'ABC' with each pen, write 'ABC' on each notebook, and roll each Hot Wheel down a runway.

3.4. Incentives

The fundamental issue in this experiment is choice of algorithm and how well the subjects execute the selected algorithm is secondary. Since our paper tests the choice of algorithm and not the choice of objects, by the logic of choice mechanisms, ideally we should give the subjects algorithms to take home. But, as this would not be a reward to most students, our approach to incentives was to show that the results are robust to variations in the incentives.

The design and incentives must consider the fact that ordering is tedious. In reviewing a previous version of the experiment for which the subjects had to order 5, 10, 15, 20, and then 25 pens, we observed that some subjects had a very large number of inconsistencies.

In this experiment, the number of sets of objects the subjects had to order was reduced to three and the largest set had less than 20 objects. When asked, subjects indicated that they were more than adequately compensated for the task as the total time involved was less than 15 min and subjects were paid over US$ 5 in cash and goods. The fact that all of our incentive systems had a flat fee component was to ensure the subjects were easy to recruit and felt adequately compensated. In the table below a flat fee of US$ 4 is indicated by F4.

We considered two incentives to increase the consistency of the orderings. If the subjects were completely consistent, they would be ordering the objects with consistent preferences. For groups 3 and 6–7, subjects were paid US$ 0.25 for consistency, indicated by 25↑, in each binary comparison and for group 4 subjects were deducted US$ 0.25 for each inconsistency, indicated by 25↓. A colleague suggested that we make the experiment incentive compatible with the secondary objective by randomly choosing two objects in the ordering and giving the subjects the object with the higher position in the ordering. We employed this incentive for the 2nd and 3rd orderings in all groups with the letters 'Ra' in the incentive table below.

Both these incentives have positive and negative features. Because both of these incentives emphasis binary choice, they may not be neutral among the class of possible algorithms. The random award of an object from the ordering is incentive compatible with the secondary objective, but may not be very salient. Specific payment for consistency is probably more salient, but might induce the subjects to modify their preferences.

For the pen orderings we tried seven variations in the design to test how the variation affected the ordering result. For the notebooks we tried one incentive system. For the Hot Wheels we used the same incentives with and without prior evaluation in which subjects developed evaluation criteria with 10 Hot Wheels not in the experiment prior to the experiment. A table of the incentive systems for the various groups is presented below:

Incentives for each group

Group	Incentive
1	F4
2	F4
3	F2+25↑
4	F5+25↓
5	F4+Ra
6	F4+Ra+25↑
7	F4+Ra+25↑
8	F4+Ra
9	F5+Ra
10	F5+Ra

3.5. Evaluate and place

In the previous version of the experiment about 90 percent of the subjects wrote with a new pen and placed it in the partial ordering before writing with a new pen. About three of the subjects in the early rounds wrote with all the pens before ordering any of them. Two of the subjects switched to the dominant algorithm after the first set because processing the list

became progressively tedious. The one subject who wrote with all the pens before trying to order them with each set presented to him became increasingly frustrated with trying to remember which pens he had pulled from the list. Since processing the list is possibly a quadratic process and this algorithm is less efficient, we organized the instructions so that all the subjects would shift to the dominant 'evaluate and place' algorithm in the practice round. Subjects could have used the list processing algorithm, but it was more natural to shift immediately to the 'evaluate and place' algorithm.

3.6. Additional measurements

Many subjects order pens quickly. Using the 'evaluate and place' algorithm some subjects can order 16 pens in less than a minute. We performed additional consistency tests to check how consistently the subjects were ordering the objects.

For each subgroup some pens from the second ordering were included in the set of pens given to the subject for the third ordering. This helped verify consistency in ordering between the second and third orderings. The rank position of objects passed without the subjects' knowledge from the completed ordering 2 to the third set to be ordered are displayed. Note: a higher number is preferred over a lower number.

Rank position of objects: passed from ordering 2 to third set to be ordered

Groups	Subgroup			
	1	2	3	4
1–6: objects in 2 $\Rightarrow$ 3	1, 2, 3, 4, 5, 6	2, 3, 4, 5, 6	2, 4, 6, 8	3, 6, 9
7–10: objects in 2 $\Rightarrow$ 3	2, 4, 5	2, 4, 5	3, 5, 6	3, 5, 6

where for Group 8, subgroup 2, objects in positions 2, 4, and 5 of the second ordering are passed to the third set to be ordered. Groups 3 through 10 were asked to make a binary comparison with selected pairs after completing the third ordering. Subjects did not see us select the objects and could not see the previous ordering and had to make each decision with only the information gained from writing the two 'ABC's or rolling the two Hot Wheels down the track and their previous experience.The selected pairs were:

Rank position of objects for binary comparison

Group	Subgroup	Binary comparisons
3–4	1	(12, 16), (3, 7), (11, 14), (5, 8), (13, 15), (4, 6), and (9, 10)
3–4	2	(11, 15), (2, 6), (10, 13), (4, 7), (12, 14), (3, 5), and (8, 9)
3–4	3	(10, 14), (1, 5), (9, 12), (3, 6), (11, 13), (2, 4), and (7, 8)
3–4	4	(10, 14), (1, 5), (9, 12), (3, 6), (11, 13), (2, 4), and (7, 8)
4–6	1	(9, 12), (8, 10), (6, 7)
4–6	2–4	(8, 11), (7, 9), (5, 6)
7–10	1	(5, 8), (4, 6), (2, 3)
7–10	2–3	(6, 9), (5, 7), (3, 4)
7–10	4	(7, 10), (6, 8), (4, 5)

3.7. Bubble sort

We also trained 16 subjects in subgroups of four to perform a bubble sort using Group 10 specifications. Prior to the experiment they examined Hot Wheels not used in the experiment and wrote a paragraph giving their evaluation criteria. Subjects were taught the bubble sort procedure on the practice round and then used this algorithm to order three additional sets. To keep the bubble sort as simple as possible, we had each subject start with the worst Hot Wheel and make binary comparisons up the ordering until he or she found the proper position for the Hot Wheel being inserting into the order.

Complete experimental instructions are available from the first author.

4. Observations

4.1. Basic algorithm

Given the instructions, all subjects used some variation of the 'evaluate and place' algorithm. They ranked the sets of objects in a sequential manner creating a partial ordering of the first $1, 2, 3, \ldots, k$ objects until finished. With the exception of Group 9, almost all subjects appeared to evaluate pens and notebooks at a constants rate because they wrote ABC with pens on paper, wrote ABC on notebooks, or rolled the Hot Wheel down the runway and immediately proceeded to the placement phase. After evaluating each new object, subjects, with few exceptions, thought for a few seconds before placing the object into the partial ordering.

4.2. Variations

In each group, there were usually one or more outliers whose behavior deviated from the rest. These deviations usually were mostly in how they evaluated the objects and less often in the way they placed the object. Rather than trying to create criteria for eliminating outliers, we made the groups large enough so that the results are robust to outliers.

An important variation in ordering pens is how many times the subjects wrote with each pen. Forty-one of the 128 subjects in groups 1–4 wrote 'ABC' with each pen only once, 83 subjects made less than six rewrites and three subjects made more than 32 rewrites. Most subjects rewrote with pens occasionally when they needed to refresh their memory about a particular pen. One subject tentatively placed each new pen in the partial pen ordering and rewrote with the pen on both sides, resulting in 57 rewrites. The subject who made 87 rewrites wrote 'ABC' with each pen several times and subjectively weighed each pen in his hand. Another subject post-processed the ordering according to visual characteristics alone without rewriting with any of the pens. These subjects expended much greater effort in evaluating each pen, but their actions certainly are not a priori a nonlinear algorithm.

Most subjects processed the pens in the order that they were given. A minority processed pens in subgroups by observable characteristics. One subject created columns for

the pens of each color, for example, all the 'ABCs' for the red pens were written in one column. This procedure facilitated placing the pens in a consistent manner in each ordering.

4.3. Evaluation problems

In Group 9 many subjects ordering Hot Wheels took less time with the larger third ordering than the smaller second ordering. It appeared that subjects needed extra time to create evaluation criteria in rounds 1 and 2. For example, one subject in Group 9 rolled each Hot Wheel at least three times down the track during the first two orderings. Then, she appeared to have made up her mind and only rolled each Hot Wheel down the track once during the third ordering. She finished the third ordering in less time than the second, though the second ordering had about half as many Hot Wheels. We ran the experiment again with Hot Wheels, Group 10, where the subjects were given 10 Hot Wheels not in the experiment and asked to write a paragraph on their evaluation criterion. With prior criteria creation, subjects appeared to evaluate the Hot Wheels in the experiment at a constant rate.

From observing the subjects' object organization and hand motions we can exclude the efficient binary sort algorithms. Subjects trained to perform a bubble sort general placed the Hot Wheel to be placed over the Hot Wheel with which they were making a binary comparison. They proceeded up the partial ordering until they placed the new addition. Because in Groups 1–10 the final placement of an object in partial ordering was almost always within one object of the initial placement we can exclude a bubble sort because the time to perform a binary comparison is clearly observable. *Our maintained hypothesis is that the subjects were using a modified bucket sort.*

Subjects, when asked to describe the algorithm they used subjects, described the criterion they used rather than how they placed the objects. The one subject who participated in the experiment and was later one of the new subjects who was trained to perform a bubble sort was emphatic that she was initially using a bucket sort.

The major difference in the human bucket sort and a computer bucket sort is the creation of the buckets. In a computer bucket sort the buckets are created before the sort is initiated. However, in the human bucket sort for a small number of objects, the subjects create buckets in a *single* motion as needed. If the subject decides that a pen should be placed between the current fourth and fifth pen the subject simply creates a new bucket between the fourth and fifth pen and the old fifth pen becomes the new sixth pen. If writing with a pen the subject decides that the pen is much better than the currently ordered pens the subject places the pen some distance to the right of the existing order.

Some insight into how the subjects placed the pens was obtained by the response of the first two groups to the following question:

"In placing a new pen into the ordering, I did it by

(a) assigning the pen a number as in the numerical ordering experiment and comparing with numbers of pens already ordered;

(b) assigning the pen a verbal descriptor such as "very good" and comparing the pen to the verbal descriptors of the pens already ordered;
(c) intuitively, without assigning numbers or verbal descriptors."

Of the 64 subjects asked this question, 3 responded with (a), 13 responded with (b), and the remaining 48 responded with (c). Thus, they were acting as if they were assigning utility values.

5. Analysis of algorithms

Before considering the regressions, it is desirable to discuss the relationship between asymptotic complexity theory and identifying algorithms using regressions based on small samples. Because the subjects write with each pen and the number of writes plus rewrites appear linear in n, the regressions should have a linear factor. Complexity theory tells us that if the subjects were using a binary algorithm operator, we should find a higher power term to be significant in the regression. The size of the sample required to demonstrate this depends on the time to perform a binary operation. We determined that the time to perform a binary comparison is 3.2 s (report available from first author) and we are 90 percent confident that it is greater than 1.5 s. We demonstrate that the number of objects is adequate.

Let us examine the three hypotheses and the regression results concerning the relationship between time T, and the number of pens, X, where we assume ϵ_i to be distributed $N(0, \sigma_\epsilon^2)$. The maintained hypothesis, a bucket sort, is:

Maintained hypothesis: bucket sort $T_i = \beta X_i + \epsilon_i$

Group	Observation	β	σ_β	t_β
1	96	7.32	0.21	34.3
2	96	8.62	0.47	18.4
3	96	9.17	0.29	31.3
4	96	8.23	0.31	26.2
5	96	8.60	0.38	22.5
6	96	8.60	0.33	25.7
7	96	9.04	0.43	20.9
6 + 7	192	8.75	0.26	33.3
8	96	10.43	0.35	29.9
9	96	8.69	0.31	27.9
10	96	10.1	0.35	28.9

We know that the function intersects the origin because each subject began working immediately and, thus, displayed no fixed cost. To test this, we estimated Eq. (1) including an intercept term α. We failed to reject the hypothesis that $\alpha = 0$ at the 0.05 significance level. Based on this, we concluded that excluding the constant term is reasonable.

Now let us consider the first alternative hypothesis: a bubble sort, where the constant term is for the evaluation process and the quadratic term for the number of binary comparisons.

First alternative: bubble sort $T_i = \beta X_i + \gamma X_i^2 + \epsilon_i$

Group	Observation	β	σ_β	t_β	γ	σ_γ	t_γ	Power
1	96	7.33	0.82	8.9	−0.00068	0.057	−0.012	0.9(0.9)
2	96	7.72	1.80	4.3	0.065	0.126	0.52	0.9(0.75)
3	96	8.25	1.12	7.3	0.066	0.078	0.85	0.9(0.9)
4	96	7.50	1.21	6.2	0.053	0.085	0.63	0.9(0.9)
5	96	7.50	1.47	5.1	0.080	0.103	0.78	0.9(0.8)
6	96	6.43	1.27	5.1	0.156	0.088	1.77	0.9(0.68)
7	96	7.35	1.63	4.5	0.184	0.171	1.07	0.8(0.5)
6+7	192	7.73	0.84	9.2	0.084	0.065	1.28	0.9(0.9)
8	96	9.81	1.32	7.4	0.068	0.139	0.49	0.9(0.78)
9	96	10.94	1.16	9.46	−0.244	0.121	−2.02	ANA
10	96	9.71	1.32	7.33	0.043	0.139	0.31	0.9(0.7)

5.1. Groups 1–8 and 10

Because we failed to reject the hypothesis that $\gamma = 0$ at the 0.05 significance level for all groups except 9 (significantly negative), we consider it reasonable to assume that $\gamma = 0$ in these cases.

5.2. Group 9

The γ coefficient is negative significant indicating that the subjects speeded up in the third round. Assuming that the subjects had trouble establishing criteria to evaluate the Hot Wheels and then speeded up once they had decided how to evaluate them, we redid the Hot Wheels experiment. Before starting the experiment the subjects were asked to run 10 Hot Wheels, not in the sets to be ordered, down the track and write a paragraph describing the criteria they would use to order the sets of Hot Wheels. With prior evaluation the results for Group 10 show no significant speedup and we can assume $\gamma = 0$.

5.3. Group 6

The one group that is nearly significant is Group 6 for which the γ coefficient would be significant at 0.08. As there was no observable change in the algorithm used by the subjects, we decided to obtain more observations to see if γ would become significant at 0.05. Combining the observations of Groups 6 with 7 results in a γ coefficient would not be significant for any $\alpha < 0.20$.

5.4. Power

To determine the power of our test that the subjects were using a modified bucket sort and not a modified bubble sort or an efficient binary sort we tested the following hypotheses:

Power of the test

Bubble vs. bucket γ for X^2	Efficient binary vs. bucket γ for $X \log_2 X$
$H_O : \gamma = 0$	$H_O : \gamma = 0$
$H_A : \gamma = b/8$	$H_A : \gamma = b$

H_O is the hypothesis that the subjects were using a modified bucket sort and H_A is the hypothesis that the subjects were using a modified bubble sort or an efficient binary sort. On average for the bubble sort, assuming the pens are in random order, each subject starts at the center, and will have to make $m/4$ binary comparisons to place the new pen in the m pens that already have been ordered. Using the Gauss formula for summing $1+2+3+\cdots+m = m(m+1)/2$ the coefficient of the quadratic term X^2 should be $b/8$, where b is the time to make a binary comparison.

For all results except Group 9 the power of the test is >0.8. With our worst case scenario for the coefficient of a modified bubble sort, the power of the test, shown in above Table, is still greater than 0.7 for these cases except Groups 6 and 7. For Group 6 combined with Group 7, the worse case scenario power of the test is >0.9.

5.5. Actual bubble sort

The results of the bubble sort regression where subjects were trained to perform a bubble sort are:

Actual bubble sort

Group	Observation	β	σ_β	$t_{\beta(96)}$	γ	σ_γ	$t_{\gamma(96)}$
Bubble	48	7.95	2.68	3.0	0.58	0.28	2.1

It is important to note that for subjects performing a bubble sort the linear evaluation process is reflected in a significant linear coefficient and the quadratic number of binary comparisons is reflected in a significant quadratic coefficient. Therefore, for all groups we can reject the hypothesis that the subjects were using a modified bubble sort.

The prospect for a simple unknown efficient binary comparison with the hand motions exhibited by the subjects is extremely remote. Nevertheless, let us consider the second alternative: an efficient binary sort, where the evaluation process is:

Second alternative: efficient binary sort $T_i = \beta X_i + \gamma X_i \log_2$

Group	Observation	β	σ_β	t_β	γ	σ_γ	t_γ	Power
1	96	6.9	1.5	4.8	0.11	0.39	0.26	0.9(0.9)
2	96	6.3	3.2	2.0	0.63	0.85	0.738	0.9(0.9)
3	96	7.7	2.0	3.9	0.40	0.53	0.76	0.9(0.9)
4	96	6.9	2.1	3.2	0.36	0.57	0.63	0.9(0.9)
5	96	6.9	2.6	2.6	0.47	0.71	0.68	0.9(0.9)
6	96	5.1	2.2	2.3	0.94	0.60	1.56	0.9(0.8)
7	96	6.4	2.8	2.3	0.83	0.88	0.949	0.9(0.75)
6+7	192	6.8	1.6	4.4	0.55	0.44	1.26	0.9(0.9)
8	96	9.7	2.3	4.3	0.25	0.71	0.35	0.9(0.9)
9	96	12.6	1.96	6.4	−1.24	0.62	−2.0	NA
10	96	9.40	2.25	4.2	0.22	0.71	0.32	0.9(0.9)

Because for the efficient binary sort we failed to reject the hypothesis that $\gamma = 0$ at the 0.05 significance level for any of the groups except Group 8 (significantly negative), we consider it reasonable to assume that $\gamma = 0$ in all these cases. Nevertheless, for completeness sake, we performed the statistical analysis to show that the subjects were not using some unknown efficient binary comparison sort. The power of the tests in the worst case scenario are greater than 0.75 in all relevant cases. Therefore, for all groups we can reject the hypothesis that the subjects were using an efficient binary comparison algorithm.

Our tests indicate that only a linear function of the number of pens provides a significant explanation for the variance in the sorting times provided the subjects have established their ranking criteria prior to starting the ordering experiment. If subjects had used any algorithm based on binary comparison, a function equal or greater than the $n \log_2 n$ should have proved significant. This means that T can be considered a linear function of X and that we can reasonably conclude that our subjects used a linear algorithm when sorting the pens, proving our hypothesis that humans act as if assigning ordinal utility values to items in sets. This hypothesis appears robust to minor changes in the incentives.

6. Consistency

Even if the variation in incentives did not affect the choice of sorting algorithm, this variation could still affect the performance of the subjects in consistently ordering the sets of objects. The design of the experiment made creating and testing consistency hypotheses using ANOVA analysis possible. We shall only show one result. The other results are available on request.

One result that clearly shows that subjects were trying is that the percent inconsistencies falls further apart in the rank position are the objects selected for testing. We performed an ANOVA analysis of the percent consistencies for 1, 2, and 3 positions apart for Group 5 (pens with random), Group 8 (notebooks), Group 9 (Hot Wheels), and Group 10 (Hot Wheels with prior criteria creation).

Percent inconsistencies for Groups 5, 8, 9, 10

Group	Observation	3 position apart	2 position apart	1 position apart
5	32	19	31	48
8	32	7.8	16	22
9	32	13	16	23
10	32	7.8	4.7	16
SIG(5, 8, 9, 10)		0.0002	0.0014	0.089

The differences in the means are significant for 1 and 2 positions apart at the 1 percent level and for 3 positions apart at the 8 percent level.

We also created four consistency statistics and used ANOVA to compare (1) random placement; (2) subjects' placement under flat incentives; and (3) subjects' placement under flat incentives with consistency tests. As would be expected, (3) induces more consistent behavior than (2), which in turn leads to more consistent behavior than (1).

7. Concluding remarks

Would the human ordering algorithm remain linear for large samples of pens? This may not be testable because subjects appear to be near their limits to meaningful discrimination among the number of pens considered in this experiment. Also, there are not really a very large number of different pens in the marketplace. Humans might linearly sort a large number of pens into equivalence classes, but the inconsistencies of the current experiment suggest that the boundaries of these equivalence classes might be fuzzy.

References

Aho, A.V., Hopcroft, J.E., Ullman, J.D., 1974. The Design and Analysis of Computer Algorithms. Addison-Wesley, Reading, MA.

Knuth, D.E., 1973. The Art of Computer Programming, Second Edition, Vol. 3. Sorting and Searching. Addison-Wesley, Reading, MA.

Myers, J., Alpert, M., 1968. Determinant buying attitudes: meaning and measurement. J. Marketing 32, 13–20.

Mehlhorn, K., 1994. Data Structures and Algorithms 1: Sorting and Searching. Springer, New York.

Payne, J., Bettman, J., Johnson, E., 1993. The Adaptive Decision Maker. Cambridge University Press, New York.

Tversky, A., 1969. Intransitivity of preferenes. Psychological Review 76 (1), 31–48.