Modularity and Team Size in Open Content Experiments

Benjamin Chiao {bchiao@umich.edu}[1]
Peking University and University of Michigan 2010/1/9
(Preliminary. The most updated version is available at
http://benjaminchiao.org/papers/opencontentsexp.pdf)

Abstract: This paper reports experimental results on open contents. Large teams were associated with a higher free-riding level than small teams, and free-riding was more severe when large teams work in a non-modular production environment. Free-riding resulted in the removal of a signaling function of price for the difficulty levels of tasks. However, this removal was not sufficient to lead to the catastrophic outcome of zero payoff. Efficiency was higher in the modular production irrespective of team size. Small teams were more efficient in the non-modular production. We have not directly investigated what prevented the catastrophic outcomes but spillover was significantly higher in the non-modular production. Here are some implications for managers concerning efficiency: if the production function is restricted to be non-modular, reduce the team size; with no such restriction, always choose the modular production function.

[1] I gratefully acknowledge the advice from Yan Chen, Michael Cohen, Jessica Litman, Jeffrey MacKie-Mason, and Andrew Schotter. I also benefited from the comments by Charlie Holt, Josh Lerner, Theo Offerman, Judith Olson, Charlie Plott, Neslihan Uler, and the anonymous referees. Sam Matti provided excellent programming. The software, experiment and tests were designed with the help of Vicky Chang, Doug Halket, Ping Lin, Seung-ju Paik, Anwar Ruff, and Hui Xu. This research is made possible by the financial and technical support from the Center for Experimental Social Science at New York University, the School of Information Lab at the University of Michigan and my advisors Jeffrey MacKie-Mason and Yan Chen. I also thank the participants of the following meetings: the Economic Science Association Regional Meetings in Tuscon 2004 and in Hong Kong 2006, the Socio-Technical Infrastructure for Electronic Transactions Fall 2004, the Incentive-Centered Design workshops at the University of Michigan, and seminars at the National Taiwan University, Peking University, and Tsinghua University.

1 Introduction

The open source movement has already challenged the biggest players in the computer science industry. Attempts have been made to broaden the scope to areas other than software. This paper focuses on one of these areas called open contents. Contents, such as software, music, films or books, are defined here to be open if one is free to use, reuse, and redistribute them[2]. How to ensure success in such area using the open source concept? Should principles prevalent in coding be also applied to contents? One principle I mainly investigate here is modularity with different team sizes.

Modularity is deeply rooted in the computer science literature[3]. The concept of information hiding, pioneered by Parnas [1972], says that non-modularity is not optimal if the inner workings of a module overlap with the responsibility of another module. For instance, non-modularity implies that errors are serially correlated across modules. Various experimental studies (e.g. see Camerer [2003] pp. 383 for a review) show that it is easier to achieve a socially inefficient outcomes (even though more socially efficient outcomes are also Nash) as team size increases when non-modularity takes the form of minimum effort games[4,5]. All these are consistent with the following theories in the open source context. Baldwin and Clark [2006] (pp. 1126) show theoretically that social efficiency decreases because non-modularity leads to a higher free-riding level among open source developers. Varying team size and modularity, Johnson [2002] (pp. 658) shows theoretically that non-modularity is socially inefficient (due to free-riding) only for large teams of developers.

[2]This is from the Open Knowledge Definition website, which claims this to be the simplest version of their more complete definition available at: http://www.opendefinition.org/, visited December 15, 2009. Other definitions exist, for example, see Newmarch [2000] and Liang [2004]. There are a few related concepts such as user-generated contents (Krumm et al. [2008]), open innovation (Chesbrough [2004]), user innovation (Von Hippel and Von Krogh [2003]) and the like. A comparison between these concepts is outside the scope of this research.

[3]Simon [1962] and Alexander [1964] are some of the earliest works.

[4]A minimum game takes this form. Individual i chooses an effort level from the set of non-negative integers to maximize his payoff determined by the minimum of the effort levels of all team members: $\text{Payoff}_i = \min_{i=1 \text{ to } N}\{\text{effort}_i\} - \text{cost}_i \times \text{effort}_i$. There are multiple equilibria in this game. Anderson et al. [2001] offers a review of the experimental results and plausible theories.

[5]This is a particularly important class of games (Camerer [2003], pp. 376-7).

However, both Baldwin and Clark [2006] and Johnson [2002] assume that the costs of development are independent across individuals, implying that the spillover of know-how does not matter in cost saving. This assumption is perhaps a key deviation from the essential features of open contents. In practice, individuals share with each other code or methods used. This paper relaxes this assumption by allowing some spillover of know-hows. Meanwhile, the game settings here are also quite different[6]. This is because this paper does not aim to directly test the above theories in an exact environment. I only report the results on similar considerations from a laboratory setting that perhaps captures more external validity than those theoretical studies, and that allows us to vary team size and the degree of modularity in a tractable manner.

Section 2 presents the experiment. Section 3 details the experimental procedures. Hypotheses are presented in Section 4. An empirical analysis is presented in Section 5. Section 6 concludes. Appedix 1 reviews the strategies for the game the subjects were asked to solve. Appendix 2 contains the instructions to subjects.

2 Experiment

The experiment is a factorial design with modularity and team size as factors. An open content production process is simulated based on a popular board game called MASTERMIND[7]. The next subsection explains the rules of MASTERMIND and provides justifications about using it for open content experiments. Appendix 1 reviews the strategies that solve the traditional (standalone) MASTERMIND. The second subsection presents the experimental design

[6]For example, the above theoretical works only study simultaneous games with restricted cost structures while this experiment allows for sequential games without such restrictions. One tradeoff is that the game is too complicated to solve. That said, I will not provide a Nash nor claim that such equilibrium exists.

[7]MASTERMIND is a registered trademark of Hasbro International Inc. It is not uncommon to adopt ready-made games in experiments. For instance, Andreoni and Varian [1999] ran an experiment using a card game a few years ago.

2.1 Open Contents and MASTERMIND

The rules of MASTERMIND are simple. In a popular version of MASTERMIND, there are pegs in 6 colors used by a combination–breaker and pegs in black and white used by the combination–setter. For each game, a combination of 4 color pegs is set secretly by the combination–setter so the search space is 6^4. The combination–breaker's task is to guess the secret combination. For each guess, the combination–setter uses black and/or white pegs to give hints to the combination–breaker. A black peg means that the color and position of a guessed peg are correct; a white peg means that the color of a guessed peg is correct but not the position.

Mathematically, the secret combination is defined as $S = \{s_1, s_2, ..., s_N\}$, where s_n is the n^{th} peg with color s, $s \in [1, k]$, $k =$ the number of colors, and $N =$ the length of the combination or the number of slots.

The t^{th} guess is defined as $h^t = \{h_1^t, h_2^t, ..., h_N^t\}$, where h_n^t is the n^{th} peg with color h, $h \in [1, k]$. $H = \{h^1, ..., h^t\}$ is a history of guesses[8].

H in MASTERMIND is a reasonable proxy for open contents in a sense that both H and many examples of open contents contain an algorithm[9] that is publicly observable, and H is free to use, reuse, and distribute in the experiment.

2.2 Experimental Design

This experiment retained the rules of MASTERMIND except that at any time subjects could post unfinished games to the public pool where everyone could see the complete history of moves of the posted games. Each game posted must be accompanied by a possibly different non-negative commission price, chosen by the poster. The one who solved it first got the commission price transferred from the reward of the poster to this person. Anyone who solved a posted game must clone the posted game first so a copy of the history of work already done would be displayed on the screen for this person to

[8]Each hint is the pair of numbers $b(h^t, S)$ and $w(h^t, S)$, that is, the number of black and white pegs. $b(h^t, S)$ is the number of subscripts n such that $s_n = h_n$. $w(h^t, S) = [\max_p b(h^t, p)] - b(h^t, S)$, where p is a vector from the set of vectors containing all permutations of the S vector. See Chvatal [1983].

[9]By categorizing the works of Alan Turing, Kurt Godel, and Alonzo Church (1930's) on Turing Machines and Recursive Functional Theory, Dennett [1995] listed these features of an algorithm: 1. Substrate neutral, only logic matters. 2. Straightforward recipe. 3. Deliver wanted results every time.

continue the work. The exact number of clones cannot be observed by the participants. Everyone was allotted three games, each with a random secret combination, in each period. There was a $1 potential reward for either solving a game by oneself or having other people to solve it through the public pool.

The experiment is a two-by-two factorial design with production i and team size j as factors where i =modularity (M) or non-modularity (NM) and j =large (L) or small (S). In the modular production, the potential reward was immediately credited to the subjects' earnings. In the non-modular production, the potential reward was credited only if all allotted games for every subject were solved–this is also called the O-ring requirement[10]. In other words, the total production (or payoff) of a team was zero if either one of its members failed in the NM production.

In each session, there were 10 periods, each of which was seven minutes long. Each period ended when the time limit was reached.

2.3 Experimental Procedures

The experiment was conducted on networked computers with human subjects in the School of Information Lab at the University of Michigan. The author developed a software that is a collaborative version of MASTERMIND. There were 16 subjects for the large teams and 4 subjects for the small teams. A total of 88 students were recruited[11]. They were mostly undergraduate students at the University of Michigan. Students already subscribed to the mailing lists of the labs received notice of the experiment. The interested students then signed up for the experiment through an online recruitment system on a first-come-first-served basis. Each session lasted for two hours. Almost all students finished the instructions in half an hour. A quiz was administered before the experiment began. The experimenter went to the carrels to check the answers. If there was a mistake, the experimenter explained to him or her individually. Individual anonymity was maintained through out the ex-

[10] It is so named, also by Kremer [1993], to capture the idea that a very insignificant part of a system can cause a complete failure. This is exemplified by the explosion of the space shuttle *Challenger* in 1986, which was caused by the failure of the O–rings.

[11] The size of the large team was chosen to accommodate the capacity of the lab, and the size of the small teams was chosen to allow greater separation of two sizes in case less than 16 subjects showed up. A size of less than 4 seemed to be too small and anonymity could become a big issue.

Team Size	Production	Number of Subjects Per Session	Number of Sessions
Large	NM	16	2
Large	M	16	2
Small	NM	4	3
Small	M	4	3

Figure 1: Features of Experimental Design

periment; subjects did not know the real world identities of the players in the software. There was a two-minute trial period before Period 1. The average payoff without the show-up fee was $15.5[12]. The features of the whole experiment is summarized in Figure 1[13].

3 Hypotheses

The general null hypothesis is no difference. The following hypotheses are the alternative hypotheses.

In both Baldwin and Clark [2006] and Johnson [2002], an individual free-rides if he or she does not complete a task while some other individuals do.

[12] Due to the nature of the non-modular production, students could end up getting nothing but the show-up fee. The show-up fee was adjusted upwards for all subjects in the same session after the experiment if there were subjects getting particularly low payments; this had not be announced during the experiment. The adjustments were made such that the total payoff was around $20.

[13] We also conducted some with-subject pilot sessions, which together with the 2-by-2 design here would complete a Solomon four-group design. Statistical tests for Solomon four-group design are known to be tough. Not until the recent decade do we find perhaps more successful methods such as van Engelenburg [1999]: "Although the [Solomon four-group] design has been proposed half a century ago, no proper data analysis techniques have been available. In this paper, it is described how data from the Solomon four-group design can be properly analyzed using maximum likelihood regression analysis." Unfortunately, I cannot rely on this method because it turned out that the data violate the normality assumption required by the maximum likelihood regression. One could also perform other statistical tests but none of these are directly related to the hypotheses here or do they fully utilize all the available data in the design.

In our experiment, this is analogous to this: A subject free-rides if he or she sets a commission price of zero in a posted game. The following hypothesis is formulated according to the above analogy[14]:

Hypothesis 1 *Free-riding in the modular production is less than that in the non-modular production.*

Issac and Walker [1988] and Issac et al. [1994] find that group size does not matter when the cost of contribution is low but group size increases contribution at a decreasing rate when the cost of contribution is high. Andreoni [2007] shows that as group size increases, the average contribution will decline. This is not because of increased free-riding but congested altruism changes the value of the social surplus to the contributor[15]. However, (Kagel et al. [1995], pp. 151-3) claim that most economists expect group size to be positively related to free-riding. It seems important to test the effects of group size against such "common sense" of economists:

Hypothesis 2 *Free-riding is less for small teams than that in large teams.*

In Johnson [2002], when the number of developers exceeds a threshold, modular software is more likely to be completed. Else, non-modular software is more likely to be completed. This is because "nonmodularity will sometimes temper [with] free-riding" (Johnson [2002], pp. 660). The next two hypotheses are consistent with this.

Hypothesis 3 *Free-riding is less (more) in non-modular than modular production for small (large) teams.*

[14] I will show in the results section that the statistical results are quite robust to different measures of free-riding.

[15] Andreoni [2007] models contribution to groups as a congestible public good. The congestion is in the hearts of the altruists rather than in the technology of the public good. Assume that Π_0 is the total social surplus generated for others by the dollars forfeited. Let $\pi_0 = \Pi_0/n$ be the average surplus for others. For a non-congested good, an individual contributor views the contribution, g, as being the total social surplus, independent of n. Then $g_0 = \Pi_0 = n\pi_0$. For a congested good, an individual contributor views the contribution, g, as being the average social surplus, independent of n. Then $g_1 = \Pi_0/n = \pi_0$. Assuming that the actual behavior might be a mixture of both, captured by $g = g_0^b g_1^{1-b} = n^b \pi_0$, he estimates that b is 0.68. That is, as group size increases, altruism of the givers is congested and the value of the contribution to the giver does not grow proportionately with the social value of the public good.

Hypothesis 4 *Efficiency is higher (lower) in non-modular than modular production for small (large) teams.*

Consistent with Parnas [1972], the following seems to be consistent with the computer scientists' expectations:

Hypothesis 5 *Efficiency is higher in the modular production than in the non-modular production.*

If one believes that this experiment captures some salient features of the minimum effort games, the following hypothesis seems reasonable because it is hard to coordinate with more team members to reach the more efficient equilibrium in such games (see Camerer [2003], pp. 383).

Hypothesis 6 *In the non-modular production, efficiency is higher for small teams.*

In contrast to Baldwin and Clark [2006] and Johnson [2002], this paper allows some spillover of know-hows. A question is whether subjects would behave such that the usual behavior in the minimum effort games no longer hold. That is, some people choose a rather high effort level in the current period such that their knowledge can be learned by others to solve games in future periods. Consistent with this suspicion, the next hypothesis is formulated as:

Hypothesis 7 *Spillover is higher in the non-modular production.*

4 Results

There are three dependent variables: free-riding, efficiency and spillover.

Free-riding is measured using three methods. Throughout the analysis, I stick with the first method and list the discrepancies whenever it applies. The first method measures the percentage of posted games set at zero commission with respect to all allotted games. The second method uses the percentage of posted games set at zero commission with respect to all posted games. The third method measures free-riding the same as the first method except that I only count those posted games, set a zero commission, that have been solved by a person other than the poster. The last method is used to capture the supply of and the demand for free-riding.

Efficiency is measured as the percentage of profits made over maximum profits. I also use a second measure, the percentage of completed games over all allotted games. These two measures coincide in the modular production but differ in the non-modular production. If only one game were not solved in the non-modular production, the first measure gives zero efficiency but the second gives positive efficiency.

Spillover is measured as the percentage of posted games over total allotted games. One can argue that for knowledge to be spread, it is necessary that a game must be posted first. In this sense, unless a game spreads negative know-hows, spillover should be positively related to the number of posted games.

In Figure 2, I group all data into four cells varying modularity and team size. The plots only use the first measures; the statistical analyses will test other measures as well.

Here are some casual visual inspection results. Hypothesis 1 seems somewhat true especially for latter periods and for large teams. Hypothesis 2 seems somewhat true especially for latter periods in the non-modular production. Hypothesis 3 seems not true as small teams seem free-ride more in the non-modular production than in the modular production. The rest of Hypothesis 3 seems true. Hypothesis 4 seems not true as small teams seem less efficient in the non-modular than in the modular production. The rest of Hypothesis 4 seems true. Hypotheses 5, 6 and 7 seem true.

In the statistical analyses, I first used an ANOVA. Standard ANOVA assumes independence, normality, and homoskedasticity of errors terms. The data do not satisfy the last two. I still performed ANOVA on these data because Conover and Iman [1980] suggest that ANOVA can still be performed the usual way after rank transforming the dependent variables.

The k^{th} observation in cell (i, j) is specified as follows:

$$\begin{aligned} \text{Dependent Variable}_{ijk} &= \mu + \text{Team Size}_i + \text{Modularity}_j \\ &\quad + \text{Team Size}_i \times \text{Modularity}_j + \epsilon_{ijk} \end{aligned} \tag{1}$$

Here are the main results using the Tukey-Kramer test for pairwise comparisons at the 5% significance level (please refer to Figure 3 for test statistics and critical values):

Result 1 *There is significantly less free-riding for small teams than that in large teams.*

Figure 2: Percentages of Free-riding, Spillover, and Efficiency Varying Modularity and Team Size

Variable	Measure	Held Constant[1]	Means Comparison Between	Means Difference	Test Statistics
Free-riding	First	S	NM & M	0.00	0.00
		L	NM & M	42.45	6.05*
		NM	S & L	-91.05	14.28*
		M	S & L	-48.80	7.65*
	Second	S	NM & M	0.00	0.00
		L	NM & M	34.60	4.74*
		NM	S & L	-88.50	13.30*
		M	S & L	-53.90	8.10*
	Third	S	NM & M	0.00	0.00
		L	NM & M	60.30	8.87*
		NM	S & L	-85.55	13.79*
		M	S & L	-25.25	4.07*
Efficiency	First	S	NM & M	-21.33	4.07*
		L	NM & M	-70.20	10.93*
		NM	S & L	61.43	10.47*
		M	S & L	12.57	2.14
	Second	S	NM & M	5.10	0.90
		L	NM & M	21.95	3.16*
		NM	S & L	10.03	1.58
		M	S & L	26.88	4.24*
Spillover	First	S	NM & M	64.63	11.24*
		L	NM & M	38.55	5.47*
		NM	S & L	-32.82	5.11*
		M	S & L	-58.90	9.16*

Note:
1. S, L, NM, and M stand for small teams, large teams, non-modularity, and modularity.
2. The critical value at the 5% significance level for a one-tailed test is 2.35. An asterisk indicates that the corresponding difference in means is significant at this level.
3. The means difference is obtained from the substraction of the first group mean by the second group mean in the "Means Comparison Between" column.

Figure 3: Tukey-Kramer Tests of Difference in Means

Result 2 *In the non-modular production, small teams are significantly more efficient*[16]*.*

Result 3 *There is significantly more spillover in the non-modular production.*

Result 4 *There is significantly less free-riding in the modular production than that in the non-modular production only for large teams.*

Result 5 *There is significantly less free-riding in the modular production than that in the non-modular production for large teams.*

Result 6 *Efficiency in the modular production is significantly higher than that in the non-modular production for large and small teams*[17]*.*

Result 7 *There is significantly more spillover for large teams.*

The overall results are the following. Hypotheses 1, 3, 4 are partly rejected, and hypotheses 2, 5, 6, 7 cannot be rejected. The test cannot reject a result not mentioned in all hypotheses: there was a higher percentage of spillover for large teams. To summarize, large teams were associated with a higher free-riding level than small teams. Free-riding was more severe when large teams work in the non-modular production. There was a higher percentage of spillover in the non-modular production, especially for large teams. When efficiency is measured as the percentage of profits made over maximum profits, small teams were more efficient than large teams in the non-modular production and efficiency was higher in the modular production irrespective of team size.

5 Conclusions

This paper reports experimental results on open contents, which are free to use, reuse, and distribute.

Price (both pecuniary and non-pecuniary) often drives the allocation of resources. However, it is sometimes difficult to measure the difficulty of a task, even less so by a central authority. If price were used as a measure, it can

[16] This is only true for the first measure of efficiency.
[17] This is only true for the first measure of efficiency.

only be a rather imperfect signal. This experiment documents a case in which price was at times not used to coordinate resources in open contents where production is decentralized. Free-riding, measured as zero price to helpers, was prevalent in the experiment. Large teams were associated with a higher free-riding level than small teams, and free-riding was more severe when large teams work in a non-modular production environment. Free-riding resulted in the removal of a signaling function of price for the difficulty levels of tasks. However, this removal was not sufficient to lead to the catastrophic outcome of zero payoff. In fact, for one measure of efficiency, small teams were more efficient than large teams in the non-modular production and efficiency was higher in the modular production irrespective of team size.

If the two team size levels here are a reasonable approximation to the team size levels in a given firm, here are some implications for managers concerning efficiency: if the production function is restricted to be non-modular, reduce the team size; with no such restriction, always choose the modular production function.

This paper calls for more explanations about some effects of non-modularity (across different team sizes) that offset the dis-incentives mentioned in the management science research and sub-optimality in the computer science literature. We have not directly investigated what prevented the catastrophic outcomes and whether subjects use the contents instead of prices to (at least implicitly) signal the difficulty levels of tasks. We also have not directly investigated whether and how open contents facilitate learning. But the data showed that spillover was significantly higher in the non-modular production. The exact mechanisms about how production costs can be lowered due to spillover might offer a promising path for future research.

References

C. Alexander. *Notes on the Synthesis of Form*. Harvard Univ Press, 1964.

S.P. Anderson, J.K. Goeree, and C.A. Holt. Minimum-effort coordination games: stochastic potential and logit equilibrium. *Games and Economic Behavior*, 34(2):177–199, 2001.

J. Andreoni. Giving gifts to groups: How altruism depends on the number of recipients. *Journal of Public Economics*, 91:1731–1749, 2007.

J. Andreoni and Hal Varian. Preplay contracting in the prisoners' dilemma. *Proceedings of the National Academy of Sciences*, 96:10933–10938, 1999.

C.Y. Baldwin and K.B. Clark. The architecture of participation: Does code architecture mitigate free riding in the open source development model? *Management Science*, 52(7):1116, 2006.

A. Bestavros and A. Belal. Mastermind, a game of diagnosis strategies. *Bulletin of the Faculty of Engineering, Alexandria University*, 1986.

C. Camerer. *Behavioral game theory: Experiments in strategic interaction*. Princeton University Press. Princeton, NJ, 2003.

H. Chesbrough. Managing open innovation. *Research-Technology Management*, 47(1):23–26, 2004.

V. Chvatal. Mastermind. *Combinatorica*, page 325 to 329, 1983.

W. J. Conover and Ronald L. Iman. Rank transformations as a bridge between parametric and nonparametric statistics. *The American Statistician*, 35(3):124–129, 1980.

D.C. Dennett. Darwin's dangerous idea. *Sciences-New York*, 35(3):34–40, 1995.

Mark Issac, James M. Walker, and Arlington W. Williams. Group size and the voluntary provision of public goods: Experimental evidence utilizing large groups. *Journal of Public Economics*, 54(1):1–36, 1994.

Mark R. Issac and James M. Walker. Group size effects in public goods provision: The voluntary contributions mechanism. *Quarterly Journal of Economics*, 103(1):179–199, 1988.

J.P. Johnson. Open source software: Private provision of a public good. *Journal of Economics & Management Strategy*, 11(4):637–662, 2002.

J.H. Kagel, A.E. Roth, and J.D. Hey. *The handbook of experimental economics*. Princeton University Press. Princeton, NJ, 1995.

Kooi. Just another mastermind strategy, 1986. Mimeo.

Kenji Koyama. An optimal mastermind strategy. *Journal of Recreational Mathematics*, 1994.

M. Kremer. The o-ring theory of economic development. *The Journal of Economic Development*, 108(3):551–575, 1993.

J. Krumm, N. Davies, and C. Narayanaswami. User-Generated Content. *IEEE Pervasive Computing*, 7(4):10–11, 2008.

L. Liang. Guide to open content licenses. *Rotterdam: Piet Zwart Institute*, 2004.

J.J. Merelo, J. Carpio, P. Castillo, V.M. Rivas, and G. Romero. Finding a needle in a haystack using hints and evolutionary computation: the case of genetic mastermind, 1999. Available from `http://geneura.ugr.es/~jmerelo/newGenMM/node3.html`.

E. Neuwirth. Some strategies for mastermind. *Zeitschrift fur Operations Research*, pages 257–278, 1982.

J. Newmarch. Open content licenses, 2000. Available from `http://jan.netcomp.monash.edu.au/opendoc/paper.html`.

D. L. Parnas. On the criteria to be used in decomposing systems into modules. *Communications of the ACM*, 15:1053–1058, 1972.

H.A. Simon. The architecture of complexity. *Proceedings of the American Philosophical Society*, pages 467–482, 1962.

Wolfgang Strobl. How to build a calculator for master minds. Available from `http://ntklotz.gmd.de/pic/mm47`. Accessed 2007/7/12.

G. van Engelenburg. Statistical Analysis for the Solomon Four-Group Design. Research Report 99-06, 1999. Faculty of Educational Science and Technology, University of Twente.

E. Von Hippel and G. Von Krogh. Open source software and the" private-collective" innovation model: Issues for organization science. *Organization Science*, 14(2):209–223, 2003.

6 Appendices

6.1 Appendix 1: Strategies for MASTERMIND

This section reviews some strategies discussed in the literature[18]. There is some indication that one can have a different assessment of difficulty level of the same posted game. This is because the number of remaining moves and time needed to find the secret combination varies, which depend on the strategy one uses below.

6.1.1 Stepwise Optimal Strategy

Each guess is optimal in the sense that it is possibly the right answer, that is, it is consistent with all guesses already played. Authors may define the term "consistency" differently. This strategy does not have an expected number of guesses, but it is guaranteed to find the solution in finite time.

A. Exhaustive Search by Koyama [1994]: In a 4 peg, 6 color game, there are $6^4 = 1296$ possible combinations. The strategy is to generate guesses with a computer in sequence and then rule out the combinations that are inconsistent with the guesses already played. For instance, if we start with AAAB, the next moves would be AAAC, AAAD... up to AAAF and then AABA, and so on. As we move along, we will be able to rule out combinations that receive no white or black pegs. The advantage of this strategy is that it runs over the search space only once.

B. Random Search by Strobl: This approach is similar to Koyama [1994] except that it generates combinations randomly. Like the exhaustive search, there are combinations that can be ruled out before being played, but it runs over the search space more than once.

6.1.2 Analytical Strategy

Combinations that are known to be incorrect are played to reduce the search space. Examples:

Pope In a 4 peg, 4 color game, we can determine what color is used and how many times each color is used by making single-color moves in the first

[18] I borrow heavily from Kooi [1986] and Merelo et al. [1999] to complete the literature review. I thank Kooi for sending me the working paper.

three rows and using a different color each time (e.g. 1st move: AAAA; 2nd move: BBBB; 3rd move: CCCC).

This approach leaves us with five possible scenarios:

- Single-color combination (e.g. AAAA). In this case, we have solved the game by the 4^{th} move.

- 2-color combination with one color used three times and the other used once (e.g. ABBB). This scenario presents $4!/3! = 4$ possibilities. Let's say we learn that the colors are A and B (from the previous three moves). By placing A in a different position each time in the next three moves, we can determine if the combination is ABBB, BABB, BBAB, or BBBA and solve the game by the 7^{th}move.

- 2-color combination with each color used twice (e.g. AABB). There are $4!/2!2! = 6$ possible combinations. Similar to the above scenario, we can use the 4^{th}, 5^{th}, and 6^{th} moves to determine the position of A and solve the game in a maximum of 7 moves.

- 3-color combination (e.g. AABC). In this scenario, we likewise use the next three moves to determine which two slots color A is in. Let's say if color A is in slots 2 and 3, we are left with two possibilities: BAAC or CAAB. We will figure out the positions of B and C in the 7^{th} move and solve the game in 8 moves.

- 4-color combination. This is the most complicated scenario. We can determine which color is in slot 4 by making the 4^{th} move AAAB and, if necessary, the 5^{th} move CCCD.WLOG, let's say color D is in slot 4. Then, in the 6^{th} move we will use the combination, AABD to get either 1, 2, or 3 positional matches. One match means A is in slot 3 and thus leaves us with two possibilities: BCAD or CBAD; Two matches indicates that the combination is either ABCD or BACD; and three matches means it is either ACBD or CABD. In each case, we will be able to identify the positions of the two remaining colors in the 7^{th} move and solve the game in a maximum of 8 moves.

Knuth (1997) This is the first paper published on solving MASTERMIND. Knuth's approach allows us to identify the combination after four moves. The expected number of moves is 4.478. The strategy is to choose a guess that minimizes the maximum number of remaining possibilities at each stage. If several guesses satisfy this condition, we will used the one that is a "valid pattern" and receives four black pegs.

Figure 4 adopts largely from a table in Koyama [1994]. It shows us the number of possible secret combinations that are consistent with the hints for each possible optimal guess. The first column lists the hints which are combinations of black and white pegs. The top row lists five possible choices for an optimal first guess.

	AAAA	AAAB	AABB	AABC	ABCD
(0,0)	625	256	256	81	16
(0,1)	0	308	256	276	152
(0,2)	0	61	96	222	312
(0,3)	0	0	16	44	136
(0,4)	0	0	1	2	9
(1,0)	500	317	256	182	108
(1,1)	0	156	208	230	252
(1,2)	0	27	36	84	132
(1,3)	0	0	0	4	8
(2,0)	150	123	114	105	96
(2,1)	0	24	32	40	48
(2,2)	0	3	4	5	6
(3,0)	20	20	20	20	20
(4,0)	1	1	1	1	1

Figure 4: Uneliminated Secret Combination Candidates Consistent with Hints Given to All Possible First Guesses

The underlined cells in Figure 4 represent the worst scenario for all possible types of first guesses. According to Knuth, AABB should be played first since it minimizes the worst scenario.

Others In Bestavros and Belal [1986], the strategy is based on information theory. The technique is to obtain as much information as possible on the secret combination with each chosen guess, be it on the average or in the worst case. With this algorithm, the secret combination can be found in 3.9 ± 0.5 or 3.8 ± 0.6 average combinations in a 4-peg and 6-color game. This strategy is an improvement on Knuth's approach.

In addition, one can also minimize the number of parts described in Kooi [1986]. Or one can minimize the entropy as in Neuwirth [1982].

6.2 Appendix 2: Instructions to Subjects

The next pages reproduce the instructions to subjects for the non-modular production only[19].

[19] The instructions for the modular production are largely except that there is no longer a group requirement. Whenever person A solves a game (or someone solves a game A has posted), the points A receive will automatically be shown as Period Earnings. The period earnings will be added to the total earnings. In the information panel, the Potential Period Earnings and Group Requirement Achieved cells are no longer there.

Experiment Instructions

Welcome. You are about to participate in an experiment that is approximately 2 hours long. You will be paid in cash at the end of the experiment. The payment will depend on the decisions you make. At various points of the instructions you will be instructed to work on short quizzes. When you have finished reading your instructions, raise your hand and the experimenter will come to check your answers before we start the actual experiment. If at anytime you have any questions, raise your hand and the experimenter will assist you. The experimenter will also entertain questions publicly when everyone has finished the instructions.

The instructions are divided into five parts.

I. Standalone Games
II. Working with Games
III. Acquiring Games
IV. Structure of the Experiment
V. How to Use the Software

I. Standalone Games

In a standalone game, there is a combination-maker and a combination-breaker. In each game, the computer will be the combination-maker and you will be the combination-breaker.

For each game, the combination-maker will randomly generate a hidden combination of four pegs using six colors (blue, red, green, yellow, cyan, pink). Colors can be used more than once. For example a combination can be (blue, blue, blue, blue), (red, cyan, blue, blue), (red, blue, green, pink), (green, pink, cyan, yellow), etc. If you have trouble seeing the colors, you can click on **Display Color Numbers** (see Figure 1 on P. 12; Area A).

You have an unlimited number of chances to guess the hidden combination. For each guess, the combination-maker will give you a hint as to how close to the hidden combination you are**. If you have a peg in the correct slot and of the correct color, a black peg will be shown. If you have a peg of the correct color but in an incorrect slot, a white peg will be shown.**

The black and white pegs for each hint refer to any of the four pegs you have chosen, not to any one specific peg. For instance, if the combination is (blue, green, red, yellow), and you guess (yellow, green, blue, cyan), then you will see one black peg and two white pegs (see Figure 2; Area A).

I.1 Reward

For each game solved, the experimenter will give out 10 points. For every 10 points you earn, you will be paid US$1.

There is a group requirement: everyone has to make sure that all **three** of his allotted games in a period are solved either by himself or by other people in this experiment. You will only receive your earnings in a period if everyone solves all his games allotted in that period. In other words, if there is one game that is not solved in a period, none of you will receive your earnings in that period.

The points you earn from solving a game will be shown in the field **Potential Period Earnings.** If **Group Requirement Achieved** becomes “True”, your **Potential Period Earnings** will be transferred to your **Period Earnings**.

Please work on Quiz 1 on P. 8-9 now. Continue with the instructions when you are done.

II. Working with Games

When you are working on a game, there are three courses of action you may take:

II.1 Completing a game
You complete a game by guessing the hidden combination correctly.

II.2 Storing a game to your Private Games Collection
You may click on **Work Later** (see Figure 2; Area B). Doing so will transfer your game, with all of the guesses you have made, into your Private Games Collection. You may resume any of the games from your Private Games Collection at any time during the same period (see Figure 4; Area A).

II.3 Posting a game
You may decide to post a game by clicking **Post** (see Figure 2; Area C). Doing so will transfer your game and all of the guesses you have made into the Public Pool (see Figure 3; Area A).

The Public Pool:

The Public Pool is a collection of games that are <u>visible to all</u> people in this experiment.

Games in the Public Pool can be <u>viewed and cloned</u> by any people in this experiment. When a person clones a game, an exact copy of the game with the history of moves and the same hidden combination is transferred to the person. This person can work on this clone and take any of the three courses of actions listed above.

Games in the Public Pool can be cloned multiple times by multiple people.

When the original game <u>or any clones</u> of it is completed, all clones and the original become void and are no longer able to be worked on.

<u>If a person clones and solves your game, you will be awarded the number of points the game was worth, just as if you finished it yourself. However, when you post a game you can choose the amount of commission you wish to pay another person for solving your game.</u> This number of points will be transferred from your account to his upon his completion of the game. For instance, if you post a game worth 10 points and offer a commission of 4 points you will receive 10 points when this game is solved by another person. Four of these points will be given to the person who solves the game as commission. Thus, you earn 6 points and the other person earns 4 points

Once your game has been posted, there is no way to control the circulation of it. People may clone many copies of your game. You can, however, grab it back from the Public Pool. Doing so removes the game from the Public Pool so that other people can no longer clone it, but it does not affect the games people have already cloned.

Since everybody is seeing the same instructions you do, everybody in your group is able to clone and post games as you are.

III. Acquiring Games

There are three ways to acquire games to work on:

III.1 Request a game from the experimenter
Each period, the experimenter allots three games for you to work on. To request one of these games, you must click **New Game** (see Figure 1; Area B). **Each one of these games you complete will earn you 10 points.**

III.2 Clone a game from the Public Pool
You may clone a game from the Public Pool to work on. Each one of these games you complete will earn you the commission amount for that game. The commission is chosen by the person who posts the game (see Figure 3; Area B).

III.3 Clone or Retrieve a Private Game
At any time you may clone or retrieve games you have stored to your Private Games Collection (see Figure 4; Area C).

Please work on Quiz 2 on P. 10 now. Continue with the instructions when you are done.

IV. Structure of the Experiment

The experiment is divided into periods. At the beginning of each period, everything except **Total Earnings** will be reset. Note that your **Period Earnings** will be added to your **Total Earnings**. Any games in the Public Pool or your Private Games Collection will be discarded.

A period ends when everyone has solved all three of his allotted games (i.e. Group Requirement Achieved becomes "true"), or in 7 minutes, whichever occurs first. The experimenter will make an announcement to remind you the remaining time when there are 2 minutes left in the period.

In this experiment, there will be 10 periods and 1 trial period.

V. How to Use the Software

V.1 Your active game
Your "Active Game" is the game in the left panel that you are working on. To make a guess, you must select a color for each of the four slots by clicking the slot and choosing the color from the drop-down menu. Once you have selected four colors, click **Check** to submit your guess and see the hint for it (see Figure 2; Area A). Note that once you have clicked **Check** you cannot change your guess.

V.2 Work later
You may store your active game to your Private Games Collection at any time by clicking **Work Later** (see Figure 2; Area B). This sends your active game to your Private Games Collection. Your Private Games Collection is the list of games under the panel titled "Your Games" on the right side of the screen (see Figure 4; Area A).

V.3 Posting
To post your active game to the public pool, click **Post**. A window titled "Post Options" will appear (see Figure 5). Here you can set the commission for this game. To the right of the window is a preview of your game as it will appear in the public pool. To cancel posting, exit the window by clicking the "X" in the upper right corner. To post the game, click **Post** (see Figure 2; Area C).

V.4 Acquiring games
There are three ways to acquire a game:

1) *New game*
   You can request one of your three allotted games from the experimenter by clicking **New Game** (see Figure 1; Area B).

2) *Retrieving a private game*
   You can retrieve a game from your Private Games Collection by looking at the list of games under the panel titled "Your Games." Here you can see the basic information of the game: Name, Guesses, and Time Added (see Figure 4; Area A).

   Name
   This is the name of the game (in number). Note that a clone will have the same name as its parent game but with an extra level number. For example, a clone of Game 1 will have the name 1.1, and a clone of Game 1.1 will have the name 1.1.1 (see Figure 4; Area B).

   Guesses
   This is the number of guesses already made in the game.

   Time Added
   This is the time you added the game into your Private Games Collection.

   Clicking on one of the games in the list will load a preview of the game on the right hand side of the screen. You may then choose **Clone** or **Retrieve** (see Figure 4; Area C). Cloning simply creates a copy of the game and leaves the original in your Private Games Collection, while retrieving it takes it out of your Private Games Collection.

3) *Retrieving a public game*
   You can browse games in the Public Pool list just as you can for those in the "Your Games" list. Here you also have all the basic information of the game. In addition to Name, Guesses, and Time Added, you also have Commission and Posted By (see Figure 3; Area A). Click on the column headers on the list to sort the games.

   Commission
   This is the number of points you earn by solving the cloned game.

Posted By
This is the User Name of the person who posts the game. No participants will know which User Name corresponds to which participant.

Click on a game to load a preview of the game on the right hand side of the screen. To clone and work on this game, click **Clone** (see Figure 3; Area B). If this game is yours, you will be able to **Grab** it out of the Public Pool. Grabbing removes the game from the Public Pool so other people will no longer be able to view and/or clone it. But this does not affect the games people already cloned.

Note: You can only acquire games when you do not have an active game. If you have an active game, you must complete it or store it to your Private Games Collection in order to work on a new game.

V.5 Information panel

You can keep track of various statistics by looking at the Information Panel on the bottom right-hand part of the screen (see Figure 1; Area C). Here is the available information:

User Name
Every person in this experiment is identified by a User Name (e.g. User 1, User 2, and so on). Your real name will be anonymous.

Potential Period Earnings
You will see an increase in your potential period earnings whenever you solve a game (or someone solves a game you post). The points you see here will be transferred to the field **Period Earnings** if **Group Requirement Achieved** becomes "True".

Period Earnings
This will be zero until everyone has solved all three of his allotted games.

Total Earnings
This is the amount of US$ you have accrued in **ALL** periods throughout the experiment. Your total earnings are simply the sum of your **Period Earnings** in dollars.

Games Left for You
This is the number of games left from the three the experimenter has allotted for you in the current period. If it reaches 0, you will not be able to request more games from the experimenter, but you will be allowed to work on public games.

Your Allotted Games Solved
This is the number of allotted games you have solved and/or the clones of your allotted games solved by others.

Group Requirement Achieved is "False" until everyone has solved three of his or her allotted games. It will become "True" when this requirement is fulfilled.

Time
This is the current time.

At the end of the experiment, the experimenter will come to each of you to record your total earnings. After that, we will shut down your screens so other students will not be able to see your earnings.

This is the end of the instructions. Please work on Quiz 3 on P.11.

Quiz 1

Please provide the hints (the black pegs and/or white pegs you expect to see) by writing B (black peg) or W (white peg) in the boxes provided.

The numbers (1-6) in the circles represent the six different colors (blue, red, green, yellow, cyan, pink) that you will see in the actual experiment. However, in the experiment, you will see both the colors and the numbers (if you click on **Display Color Numbers**).

Note: keep in mind that the position of the black and white pegs does not necessarily correspond to the positions of the guesses. Black pegs are always shown first (i.e. to the left of white pegs).

Please turn to the next page.

Example:

The hidden combination is

2 4 1 6

Your guess is:

4 5 1 2

Hint:

B W W

1. The hidden combination is

   2 3 1 5

   Your guess is:

   5 3 1 2

   Hint:

Quiz 2

Person 1 posts a game with the name, Game 2, which is worth 10 points and sets the commission to 5.

1. Person 2 clones this game. Therefore, the cloned game, Game 2.1 is created. How many points will Persons 1 and 2 each receive, if Person 2 solves Game 2.1?

   Person 1: [ ] points

   Person 2: [ ] points

2. After Person 2 has worked on Game 2.1, he however decides to repost it to the public pool and sets the commission to 3. Person 3 clones it, thereby creating Game 2.1.1. If Person 3 solves Game 2.1.1, how many points will Person 1, 2, and 3 each receive?

   Person 1: [ ] points

   Person 2: [ ] points

   Person 3: [ ] points

3. Now if Person 3 has not solved Game 2.1.1, and Person 2 is still working on Game 2.1 after he has posted it. This means that Person 2 and Person 3 are working concurrently on Game 2. If Person 2 solves Game 2.1 first, how many points will Persons 1, 2, and 3 each receive?

   Person 1: [ ] points

   Person 2: [ ] points

   Person 3: [ ] points

Quiz 3

1. Each person is allotted three games. You have solved two of your own games that are worth 10 points each, and your third game is solved by another person whom you have agreed to pay 4 points to. However, not every person in this experiment has solved all three of his games yet. What will you see in the following fields?

   Potential Period Earnings: ☐

   Period Earnings: ☐

   Games Left for You: ☐

   Your Allotted Games Solved: ☐

   Group Requirement Achieved: ☐

2. Now every person has solved three games, what do you see in the following fields?

   Potential Period Earnings: ☐

   Period Earnings: ☐

   Games Left for You: ☐

   Your Allotted Games Solved: ☐

   Group Requirement Achieved: ☐

Please raise your hand when you are done so the experimenter can come to check your answers.

Figure 1

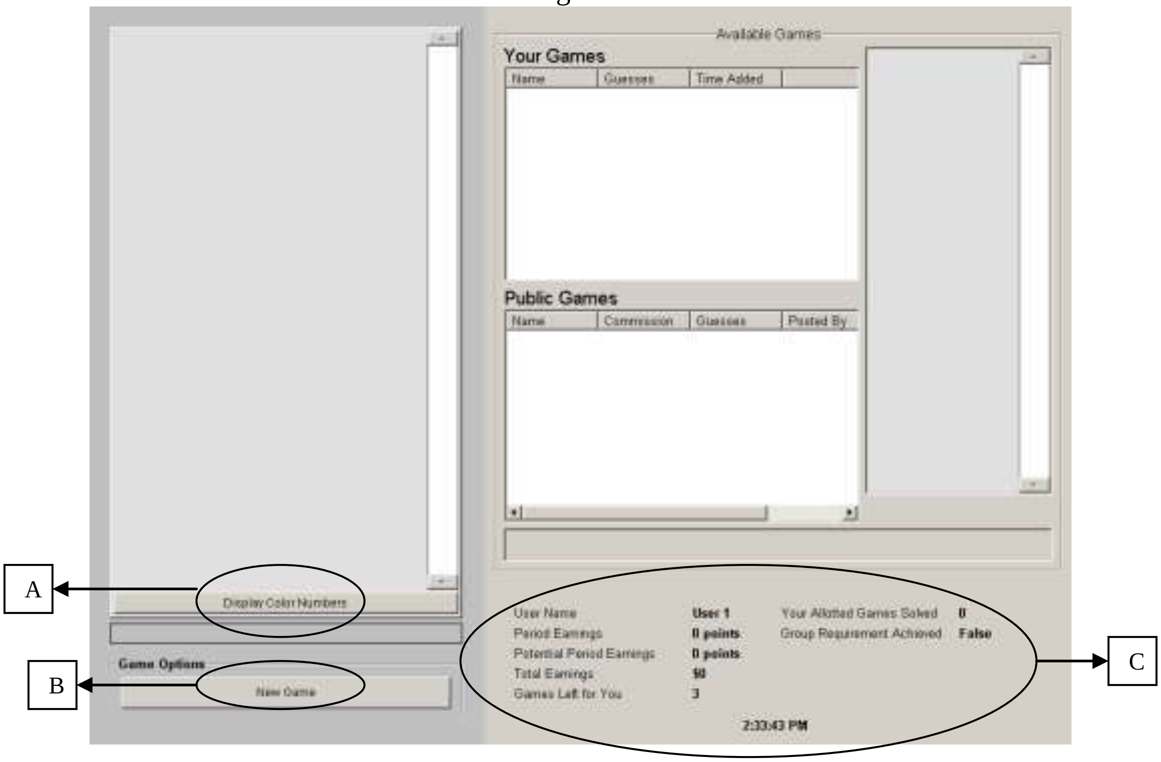


Figure 2

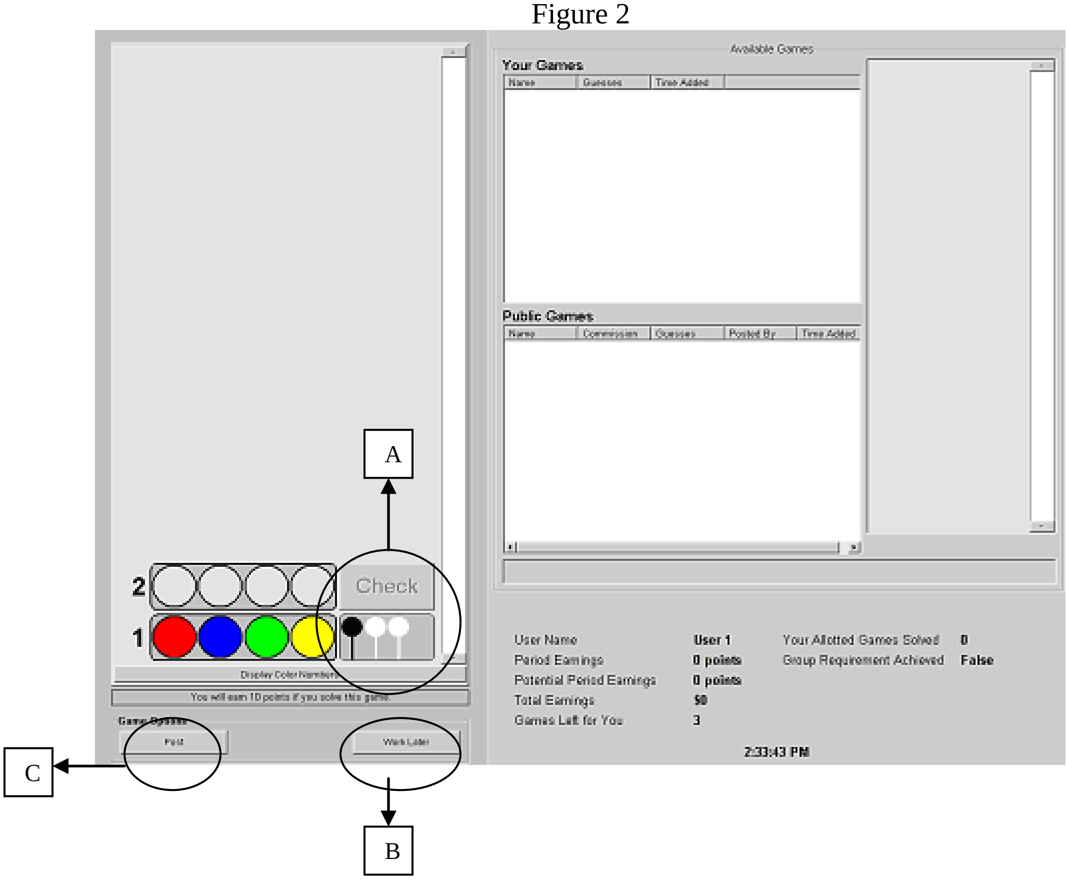


Figure 3

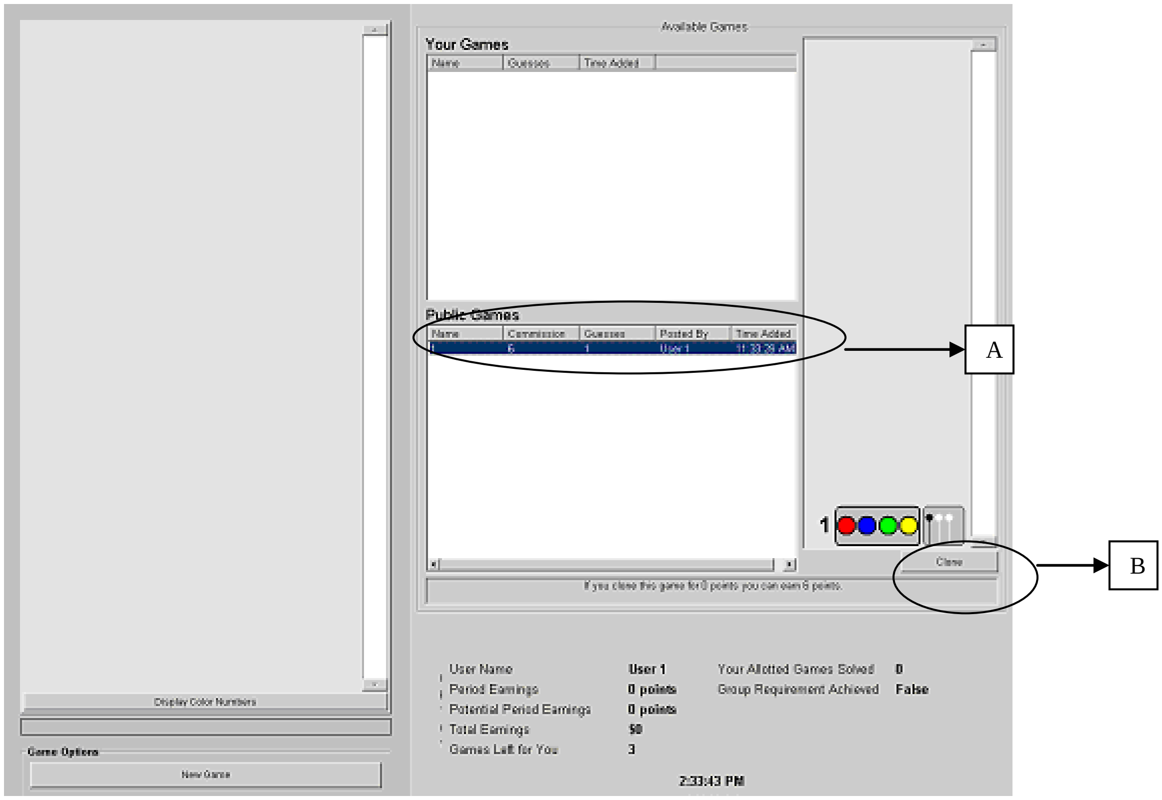

Available Games
Your Games
Public Games
A
B
Clone
Display Color Numbers
Game Options
New Game
User Name
User 1
Period Earnings
0 points
Potential Period Earnings
0 points
Total Earnings
50
Games Left for You
3
Your Allotted Games Solved
0
Group Requirement Achieved
False
2:33:43 PM

Figure 4

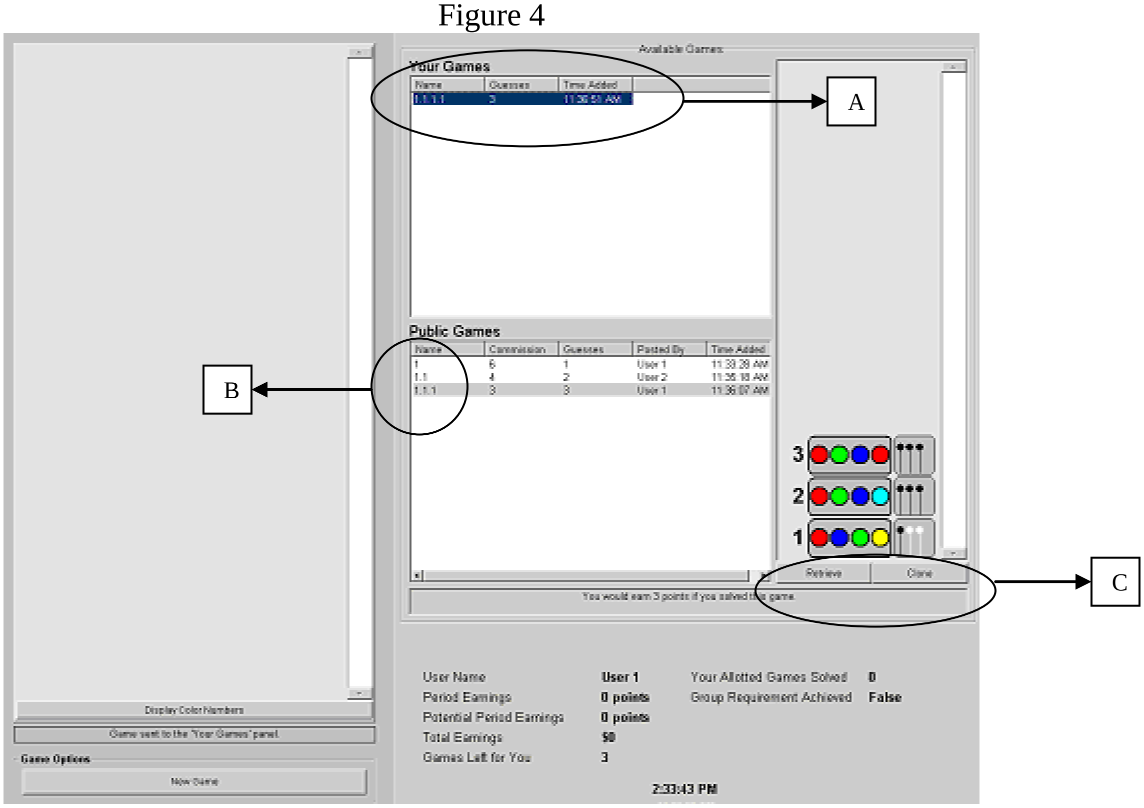
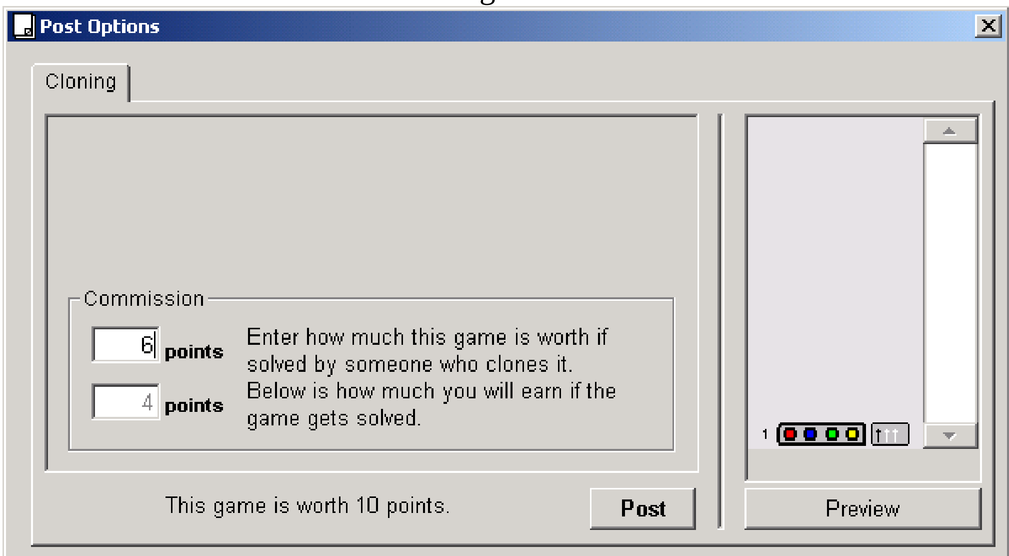

Available Games
Your Games
Public Games
A
B
C
Retrieve
Clone
Display Color Numbers
Game Options
New Game
User Name
User 1
Period Earnings
0 points
Potential Period Earnings
0 points
Total Earnings
50
Games Left for You
3
Your Allotted Games Solved
0
Group Requirement Achieved
False
2:33:43 PM

Figure 5


Post Options
Cloning
Commission
6 points
4 points
Enter how much this game is worth if solved by someone who clones it. Below is how much you will earn if the game gets solved.
This game is worth 10 points.
Post
Preview