

Rivalry and the Rules of the Game in Processes of Collective Invention

Studying Programming Contests as Microcosms of Innovation and Regulation in the Digital Economy

By

Paul A. David¹ & Matthijs den Besten²

¹*Stanford University, Ecole Polytechnique & Telecom Paris-Tech*

²*Ecole Polytechnique*

A Project Supported by the Chaire Innovation et Régulation des Services Numériques

In collaboration with

Ned Gulley

and

Karim Lakhani

The MathWorks, Inc.

Harvard Business School

This draft: January 20, 2010



This work is licensed under the Creative Commons Attribution-NonCommercial-NoDerivs 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc-nd/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.

Contact Author: P. A. David
Department of Economics
Stanford University, Stanford CA 94305
pad@stanford.edu

Rivalry and the Rules of the Game in Processes of Collective Invention: Programming Contests as a Microcosm for the Study of Innovation and Regulation in the Digital Economy

Paul A. David & Matthijs den Besten

1. Context : Open Innovation Strategies and the Revival of “Public” Contests

Recent corporate and academic interest in the emergence of “open innovation strategies” has focused upon the offering prizes to winners of public competitions to solve technical problems in process or product design, and the tensions that firms face in seeking to exploit distributed external knowledge resources by sharing with them the internally held information assets required to find satisfactory solutions to those problems. “Sharing” with outsiders who could be or are rivals may be “socially efficient” and therefore beneficial *for the ensemble* of agents (firms) forming the industry, but it entails the surrender of exclusive control of technological and market information that forms a basis on which individual participants can extract “information rents.”

An important issue for economic policy design, therefore, is to better understand the balance of these conflicting forces affecting the incentives for inventive and innovative efforts, and its effectiveness. In other words, even though it may require freely licensing existing intellectual property, there may be dynamic forms of monopolistically competitive equilibria in repeated games of open rivalry that involving reciprocated revelation of information by the players. Knowledge of others solution-approaches and progress could be sufficiently beneficial in their own problem-solving activities to motivate the competitors to reciprocate, and allow them individually to rationally anticipate greater success in arriving at future highly profitable innovations for specialty markets. Openness about the innovation process thus could be a spur to vigorous “Schumpeterian competition” (private innovative investment), and not simply an invitation to unrequited information borrowing and imitative behaviors that in the limit would dissipate profits.

2. Introduction: Programming Contests in Historical and Technical Perspective

The MatLab competition, which is the focus of our research interest here, may be viewed along with the many other computer programming competition that have proliferated rapidly in the past decade, as having developed from the fusion of two older forms of problem-solving contest: competitions among mathematicians (and students of mathematics), and engineering design competitions.¹

Historically, mathematical contests emerged in a loosely institutionalized form in Western Europe during the 16th century and pitted individuals adept in the use of the relatively novel, algebraic calculation methods (“algorithms”) against the master’s of older procedures for rapid reckoning: for example, the title page of the 1529 edition of the *Rechenbücher* of Adam Riese depicts a contest between an abacist and a algorist, but mathematical contests by that date were becoming personalized challenges, as was the case in the famous competition during the 1530’s that involved the algebraic solution of cubic equations -- in which Niccolo Tartaglia [c.1499-1557)] convincingly demonstrated his

¹ For some of this background see P. A. David and M. den Besten, “The MatLab Competition as a Paradigmatic Laboratory for Micro-Level Experimental Studies of Collective Creativity.” Version 6 (December 2009).

command of a technique far more general, and hence powerful in tackling the array of problems posed, than was the method about which Antonio Maria Fior had publicly boasted.

Those competitions, served to test publicly individuals' claims to expertise in advanced areas of renaissance mathematics and bring the winners to the attention of their peers and potential aristocratic patrons, contributed at the same time to advancing the demonstrable computational superiority of algebraic methods of solving specific classes of problems – such as finding the roots of higher-order polynomial equations. David (2008) connects the emergence of these public “exhibitions” with the origins of open science institutions, although it was true that as a rule winning contestants were not obliged to reveal the techniques that had brought them victory and, like the well-recorded incident involving Tartaglia and Girlandio Cardono, public disclosure came about in ways that they had not intended.

With the growing professionalization of mathematics as an applied career that could provide a reasonably secure livelihood in the service of the elite, and the seventeenth century transformations of patronage practices that saw the emergence of regalian and princely sponsorship of academies, and the institutionalized submission of mathematical treatises for formal peer evaluation of full mathematical proofs, as conditions for the award of appointments and university positions, public mathematical competitions at advanced level shed their early function of demonstrating individuals' possession of arcane but evidently useful knowledge. Contests of mathematical ability, however, far from being abandoned found a new purpose, and not only survived but flourished as an adjunct of formal mathematical education.

By the latter part of the nineteenth century, with the spread of secondary schooling in Europe, mathematical contests re-emerged as a increasingly popular vehicle for engaging the interests of students and identifying precocious talent. It appears that among the earliest organized competitions of this kind were those organized by certain *gymnasia* in Moscow, followed by those in Hungary, and Roumania in the 1880's, whence their promotion and the preparation of student to compete became general throughout primary schools in those countries (see, e.g, Kenderov 2008). The year 1894 saw the founding (by Dániel Arany, a high school teacher in Győr, Hungary) of the famous mathematics journal *KöMal* (an acronym of the Hungarian title of the journal, which translates as “High School Mathematics and Physics Journal”. *KöMal* invited readers to submit solutions to the problems it published, and became essential in the preparation of students and teachers for competitions that it undertook to organize for students. The format of these competitions was taken up by the school systems of other countries early in the following century. Whereas the outcome of competitions of this kind was decided on the basis of how many problems (of a specified level of originality and difficulty) could be correctly answered within a finite time duration, contests among those identified as proficient from their performance began to pose questions that called for full mathematical proofs. In 1934, a Mathematical Olympiad was organized in Leningrad (now once again St. Petersburg), and as the contestants were set problems calling for full proofs the more elevated label of “Olympiad” was employed to distinguish this form of competition from others.

During the post WWII era, this educational function came to dominate completely the purposes for which “math competitions” were organized, and they proliferated rapidly and increased in variety at the local, nationally and international levels. Eastern European and Russian schools once again took the lead in the movement to organization international contests for advanced students, with the first International Mathematics Olympiad (IMO) being launched in Romania in 1959 with local participants being joined by those from Buglaria, Czechoslovakia, the GDR, Hungary, Poland and the USSR. By 2007, when the IMO was held in Vietnam, 529 contestants from 94 countries participated, and in many instances those entering had come up through the ranks of winners of national and regional

competitions [Kenderov (2008); Wikipedia entry on “mathematics competitions”:
http://en.wikipedia.org/wiki/Mathematics_competitions].

Although student competitions at first were restricted to secondary school pupils and subsequently moved down into the primary schools, the twentieth century spread of mass university education has seen their extension into the tertiary educational system, where they are organized by individual colleges, universities and institutes, as well as sponsored by professional societies and other bodies. University scholarship math competitions for high school students in the U.S. currently draw sponsorship from an assortment of private and public sector patrons, including Department of Homeland Security, Coca-Cola and Intel. Math competitions at the university level involve far fewer institutions and contestants than those organized at earlier educational levels, but their ostensible purposes, similarly, seek to stimulate, identify and reward those who display outstanding talent by awarding scholarship prizes providing access to more advanced study. This characterizes one of the oldest and most prestigious of the college level math competitions in the U.S. -- the Putnam Competition, which accepts both individual entrants and college “teams” that are comprised of three members who work independently but have their scores pooled to determine the winning institution [http://en.wikipedia.org/wiki/William_Lowell_Putnam_Mathematical_Competition].

Yet, today comparatively few mathematics competitions are organized at the college level, and this situation stands in marked contrast with the growth in the popularity of engineering proto-type design competitions. The latter have become quite ubiquitous features of the programs of schools of engineering in the U.S. and western Europe, and in the former, at least, regularly attract the sponsorship of professional societies and both individual corporations and industry consortia. The rise and proliferation of computer programming competitions therefore might be seen at this stage to be more closely related to the latter, technologically-oriented movement.

Yet, it may also be remarked that the modern programming competition serves functions akin to those found in the early history of mathematical contests. Certainly, their concern with the development of algorithms that have superior functional attributes shares something with engineering, while the revelation of superior expertise in a technical field that has remained remarkably un-professionalized resembles in some aspects of both the early contests among mathematicians, and the interests of the state and the industrial community in mechanisms through which educational institutions could discover and cultivate a cadre of individuals who possessed extraordinary mathematical ability – which found increasingly wider practical applications. Undoubtedly, the intellectual challenges of programming competitions, as well the intrinsic satisfactions of solving novel problems, and, for some, doing so under the tension of rivalry with others, shares important qualities that motivate the contestants in the kindred pursuits of mathematics and engineering design.

Nevertheless, there also is an undeniably close affinity between programming competitions and the tradition of educationally-oriented mathematics competitions, specifically in the international olympiad for secondary school pupils. In May 1989 the first International Olympiad in Informatics (IOI) was initiated and sponsored by UNESCO, following a proposal made in 1987 by the Bulgarian delegate to the organization (a Prof. Sendov). According to the IOI’s official website: “The primary goal of the IOI is to stimulate interest in informatics (computing science) and information technology. Another important goal is to bring together exceptionally talented pupils from various countries and to have them share scientific and cultural experiences.” Thus, the purpose as well as the format of event is closely congruent with that of the international mathematical olympiads: participating countries send “teams” of four students who compete individually, seeking to maximize their score by solving algorithmic programming problems (typically on a personal computer) during two days of the competition. [See: <http://www.ioinformatics.org/history.shtml>.]

What is novel and intriguing about modern programming competitions, however, are the special characteristics of computer software as an artefact – “machines implemented by text”, possessing unusual immediacy of self-validation at both low and high levels of granularity, and almost instantaneously communicable at negligible cost via computer-mediated telecommunication networks. These characteristics of software, as will be seen, greatly facilitate the use of programming competitions as instruments for the study of invention in microscopic detail, making it feasible to develop systematic programs of experimentation with the structure of such contests to investigate social and technical factors that affect both individual and collective creativity. The exploitation of this opportunity, by a collaborative research project organized in Paris under the aegis of the *Chaire Innovation et Regulation des Services Numeriques*, is the goal toward which this background document is directed.

3. The MatLab competition: a vehicle for exploring “collaboration among rivals” and “rivalry within collaborations”

MatLab is a software environment for numerical computing that is quite popular among engineers engaged in fields such as signal processing and the modelling of fluid dynamics. Typically MatLab is used for rapid prototyping of ideas, while final products are still written in “production code” like C or FORTRAN. Due to the high level of specialization and the early stage of the product cycle associated with MatLab use, engineers employing this environment tend to remain fairly isolated. Since 1998, The Mathworks Inc. has been organizing a regular programming contest in MatLab which helps alleviate this isolation. [see <http://www.mathworks.com/contest/overview.html>].

Twice a year Mathworks informs programmers through an announcement on its online newsletter about a forthcoming contest, and invites them to submit solutions to the problem that it has set. Contest problems are defined in advance and without ambiguity by the organizers; to be accepted and evaluated as submissions, algorithmic solutions should be expressed in the MatLab programming language and executable on the MatLab programming environment. The MatLab programming contest typically defines combinatorial optimization problems that are “NP-complete” (or “NP-hard”) – a property that ensures that algorithms will not be able to find the optimal solution to the problems in reasonable time for problem instances of a decent size and so they will invariably have to resort to satisficing.

The algorithms are ranked according to the relative score they obtain on instances of the problem and the winning solution is defined as the algorithm that yields the best average score. The quality of solutions is assessed by applying the submitted programs to one or more instances of the problem. In other words, the score of the solution generated by the program is computed on the problem’s objective function and the program’s run time in the environment is measured along with other properties of the code. In most cases the rank assigned to a submission is based on the objective function score alone, but other measures, such as run time, total lines of code or the degree of cyclomatic complexity² may be given weight in the scoring to enhance the challenge.

For participating engineers and other users of MatLab, these contests provide an opportunity to show off their ability to develop algorithmic solutions to difficult problems in a short amount of time, while also offering them an opportunity to learn some MatLab tips and tricks from others. Reciprocally, the MatLab contests provide The Mathworks, Inc. an opportunity to show off the potential of their programming environment, as well as to learn

²On the cyclomatic or “McCabe” complexity metric, see http://en.wikipedia.org/wiki/Cyclomatic_complexity; higher values generally are taken to be associated with greater difficulties in understanding, and hence in de-bugging and maintaining the code.

how it is used by customers and, in turn teach its user-base new and more effective ways of exploiting the potentials of their evolving software system.

Over the years, the MathLab contests' organizers have tried to manage each event to make it as attractive as possible for their audience and participants, making changes in the rewards and rules regarding the disclosure of submissions, in an effort to make the event more entertaining and attractive for the participants – consistent with enhancing their expected collective problem-solving performance. The variety of arrangements tried in this manner represent “natural experiments” in the way that the behaviors of agents engaging in “collaborative creativity” respond to alterations in some of the institutional parameters regulating the conditions of competition and cooperation. This provides an opportunity for researchers to study at the very micro-level the ways in which rivalry can be contained and stimulated in order to improve the best performance yielded by the structured “collaboration of rivals.” It offers a microscopic window on phenomena that are present within “open science” and may well appear among firms pursuing “open strategies of innovation.”

From the research perspective, what sets the MatLab programming contest apart from many other competitions of this kind, such as the International Olympiad in Informatics (IOI), and TopCoder, is that the participants are encouraged to submit solutions early and often. Moreover, the participants are allowed – during some intervals within the week-long event – to access each other's submissions and improve upon, or otherwise utilize their code (Cormack et al., 2006). The set up of the contest thus allows for both competition and cooperation among the participants. This setting consequently can be seen to provide a generic model that captures important features of the pursuit of knowledge in academic, open science research, and industrial “innovation” processes (Dasgupta and David, 1994; David, 2003, Gulley, 2001 and 2004). This aspect of the contest has made it an attractive paradigm for the research purposes of the Chaire Innovation et Regulation, since the latter aims to gain, *inter alia* a deeper understanding on how innovation in the domain of digital services may be promoted by means of “regulation.”

In this context, regulation is about setting the rules in the game, providing a level playing field to potential entrants and promoting fair play and equal access. The goal of the regulator is to define rules that result in levels of competition and collaboration that are conducive to the highest possible rate of innovation, or direct innovations to the solution of certain classes of problems and solution-forms that have particular properties that are desirable. In the simplest of MatLab-like contests there will be a transparent scalar metric of performance, represented by the rate of improvement of the submissions' scores on the objective function. Starting from the familiar structure of the MatLab contests as a platform, it is of considerable interest to seek to experiment with changes in this environment, altering the rules of the game, the extent and timing of periods during which it was possible to see only the performance scores of the field of competitors (either wholly or only partially), and the intervals in which accessing and “borrowing” the code submitted by others was permitted. These do not exhaust the dimensions of the contest structure and its rules that can in principle be manipulated, but they do provide scope for experimental designs that would not radically transform the character of the “game” and diminish the value of the archive of past competition data in helping one understand and interpret the experimental results.

Being in a position to study the entire evolution of individual participants' submission over the course of a competition, and to trace the migration of code among members of the ensemble of contestants – tracking the successive movements of specific lines of code, and files among the submissions of identified contestants, will enable researchers working in this laboratory to empirically explore phenomena of much interest to cognitive scientists and psychologists concerned with the role of imitation in creative adaptation. They would find there a new and far higher resolution lens with which to examine the micro-dynamics of informational “spill-overs”-- about which there has been much discussion by economists but

too little solid empirical evidence. It is one thing to draw inferences about “knowledge spillovers” from patterns of citations in patents and in scientific articles, or to devise and theoretically analyze the implications of conjectured strategies of information-masking and disclosure, and quite something else to see who incorporated what block or code from which program, and what previous borrowings were carried by that block, and where the parties in these exchanges (or their latest visible submission) were positioned in the distribution of performance when each transaction occurred. Indeed, the accumulation of an archive of the histories of successive competition would permit studies of the behaviors of individuals, from which characteristic patterns reflecting different strategies and distinctive approaches to specific classes of algorithmic problems might be identified.

It is important to keep in mind that while the reputation of the Matlab competition can impart powerful impetus to the formation of such an observational and experimental platform, the separate and distinctive identity of that competition is an asset it is reasonable to expect Mathworks will be concerned to protect. The company uses these recurring contests to connect with its power users, to create a sense of community among them, and to learn more about how the MatLab environment is utilized to solve challenging computational problems. The source code submitted by these power user participants also serves as a treasure trove of examples that mainstream users can mine in search for potential solutions to their own problems. In a sense, Mathworks is constrained by the reputation and traditions that it has created in the Matlab competition as an event whose primary purposes are to entertain, to educate, and to engage and identify extraordinary programming talent in working with that particular programming environment. The experimental aspect of the research program upon which we are embarked therefore has a contribution to make by developing a distinct competition, that builds upon the experience of Matlab and forms a symbiotic relationship with that competition, while not risking the confusion of the two in the eyes and minds of the members of the two communities of participating programmers.

One also should not lose sight of the important fact that the MatLab contest is conceived as a spectator sport that a particular educated public can watch in awe at the feats that are achieved as they evolve during the contest, and which incidentally serves to promote the use of this programming environment. Popular games often involve a combination of competition and cooperation by pitting teams against each other as is the case in soccer or by allowing individuals to engage in shifting coalitions across teams as is the case in the Tour de France. A game like snooker, which involves only two players, is interesting because players continuously have to balance their chances of achieving a high score with their opponents’ chances of taken advantage of the field they have prepared. The MatLab contest combines snooker’s *stygmegic* self-revealing with the potential for the strategies and internal team politics of the peleton in the Tour de France, and therefore can be very entertaining to spectators as well as its participants.

Whether it will be necessary to sacrifice some of those qualities in order to create an experimental platform on which competitions are designed and run for research purposes is an issue that must be carefully considered. For research purposes one would also want to explore extremes of the contest’s configuration space, some of which may prove to be not so entertaining; similarly, one may wish to subject the submissions of the participants to a level of scrutiny that might be unwelcome to some or many individuals.

4. Research approaches to the MatLab contest archives: individuals and ensembles

Ned Gulley and Karim Lakhani, who are collaborating in the project undertaken by the Chaire I & R began an empirical study of the characteristics of the leading competitors

and final winners of past MathLab contests.³ Their initial examination of the respective groups of contestants who were the authors of leading entries during the course of each competition, and of the strategies followed by the individuals that emerged as having submitted the winning entry in each of these contests, has revealed the collective nature of the leading and winning entries.

The progress of individuals as well as that of the ensemble in competitions of this kind is incremental and entails the borrowing and reuse of knowledge (code), as Gulley and Lakhani's (2009) analysis of a typical winner's entry illustrates. Its 545 lines of code contained the code of 29 contestants besides the winner, who authored only 114 of its lines, a mere 12 of which were new in his last (and winning) entry. In this regard, the possibilities that software affords for research on the creation of new products and processes are remarkable and exciting. It is possible readily to trace the movement of elements of contributed code from their debut in the competition to their disappearance or survival in subsequent entries, and of thereby uncovering such regularities as may appear in the patterns of borrowing and code reuse and recombination of the various contests.⁴ This may provide more general insights into the dynamics of interactive peer rivalries in the context of cooperative, community-based knowledge creation – a class of learning processes for alternative paradigms are provided by “open science” research in academia, and the “community mode” of free and open source software production.⁵ More fine-grained econometric analysis of data tracking the activities of individual competitors through successive time intervals and distinct phases of the MatLab competition is expected to be more informative about the variety of strategies and the effects of the distinct phases of the contests varying regime of disclosure.

A different line of approach focuses less on the leaders' and ultimate winners' characteristics and strategies, and seeks instead to understand the connections between the varying regimes of information disclosure that are a unique feature of the basic MatLab competition and the behaviour of the ensemble of participants. The two most obtrusive facets of macro-level level behaviour pertain to the levels of activity exhibited by the contestants as a group, and the evolution of performance metrics of characterizing their submissions – the quality valorisation of the ensemble's “output” over the course of the competition.

4.1 The effects of the imposed disclosure regime on the ensemble's activity levels

Our preliminary empirical analysis of the data from eight contests examined the effects of alteration of disclosure regimes upon the levels of contestant's activity, as gauged alternatively by the aggregate frequency of submissions within specified time intervals, the

³ See N. Gulley and K. Lakhani, “Patterns of Innovation Generation in a Collaborative Community: Exploring the Relationship between Knowledge Novelty and Reuse.” MIT Collective Intelligence Seminar, (October) 2008; Teleconference presentation for the Chaire I&R at Orange Labs, Paris, May 2009.

⁴ Code migration and reuse can be studied in the context of open source software projects, by analysis and reuse, but the original point of blocks of codes of fine granularity is more difficult to establish without access to the complete CVS archive, because early files and whole directories are often deleted and replaced by new commits, and trawls through a large archive to match particularly identifiable code fragments require extensive computing resources.

⁵ See, e.g., on “open science,” P. A. David, M. den Besten and R. Schroeder, “Collaborative Research in e-Science and Open Access to Information,” Forthcoming in *The International Handbook of eResearch*, Edited by J. Hunsinger, M. Allen and L. Klastrup, New York and Berlin: Springer Verlag, 2009. [Preprint available as SIEPR Discussion Paper No.021 (January) 2009, at <http://siepr.stanford.edu/papers/pdf/08-21.html>.] On free and open source software production, see P. A. David and J.-M. Dalle, “It Takes All Kinds”: A Simulation Modeling Perspective on Motivation and Coordination in Libre Software Development Projects,” SIEPR Discussion Paper 07-024 (1 December 2007). [Available at: <http://siepr.stanford.edu/publicationsprofile/1785>.

distribution of average numbers entries per participant per time period, and the distribution of elapsed time between first and last entries within given time intervals.

Each of these activity metrics can be computed for the three distinct disclosure regimes in eight of the 11 MatLab contests held since 1998, those being the week-long competition that featured distinct “phases” in which different disclosure conditions obtained: (i) “darkness,” when neither the performance scores nor the code of the submitted algorithms is revealed, (ii) “twilight,” when only the score of the submissions is disclosed, and “daylight” when all both the score and the code are revealed upon submission.⁶ We find that the distributions of the levels of activity of the participants during 4-hour intervals differ among these regime-phases in rather complicated ways, suggesting there is heterogeneity in the participants’ strategies of revelation.

For example: the distributions of positive submissions (in 4 hour intervals) then exhibit higher per competitor activity levels during “daylight” than during “twilight” and this measure of activity level during “twilight” dominates that during “darkness; but the frequency of *zero* submissions per competitor during daylight during “darkness” dominates that during “daylight”, which in terms dominates that during “twilight.” Are some competitors spurred to greater activity by seeing the performance results of others, whereas for other a reluctance to let rivals see their code decreases the average frequency of their submissions? It appears that the frequency of intervals in which activity (positive submissions) per competitor reaches high levels is greatest during “daylight”, which may reflect the response of a subset among the participants.

But for others, the opportunity that “daylight” efforts to study others’ code competes for time with modifying one’s own code and therefore lowers the frequency of positive submissions per competitor compared with what is observed during phases of “twilight”. When one compares the distributions of the time elapsed between successive submissions of code *by the same author*, the modal interval length generally is longer in the “twilight” phases than is the case during “darkness.” This cannot be attributable to the distractions of studying others’ code, because the latter are not disclosed in either of those phases. But there could be a “discouragement effect” on the average activity levels of participates who learn during “twilight” that their performance scores are far behind those of the leaders, and so they may bide their time until the onset of “daylight” when they can observe what those ahead of them have been doing. What emerges quite clearly from the data explorations and statistical tests based on the data for the eight comparable contests that are reported in the Appendix is that compared to the other two regimes, daylight has marked stimulating effects on the distribution of activity levels per participant. The distribution becomes less skewed, as activity in submitting new entries becomes less concentrated in the hands of a small minority, while at the same time those the average number of entries per participant (within 4 hour intervals) is raised further among those at the upper end of the distribution.

While it appears that heightened activity among the contestants during “daylight” is accompanied by the converges of the distribution of “scores” toward those at the frontier of the ensemble’s performance, break-throughs at the frontier re-establish dispersion towards the late stage of these competitions. A second indication of the complex relationship between “activity level effects” and “performance effects” associated with different disclosure regimes is found in the divergence between the general finding that “darkness” and “twilight” do not differ significantly in their associated distributions of per participant activity, and the observation that twilight did boost activity in the case of one competition: “Splicing”. As the following section details, this competition was unusual, however, in having introduced a 24-

⁶ See P. A. David and M. den Besten, “Measuring the ‘Economy of Thoughts’ – an aggregate level analysis.” Version 5bis. I&R Programming Competitions Project Working Paper. November 2009.

hour interval of “twilight” on the penultimate day of the competition – eliciting some rather dramatic results not only a surge in activity but a marked improvement of performance scores at the frontier.

5. Strategic responses to the announced sequence of disclosure regimes: insights from the MatLab “Splicing” competition

In the week ending 14th November 2007, Mathworks ran its 17th MatLab programming contest. The objective of this contest was the development of an algorithm for gene splicing. That is, participants needed to come up with a method to transform test sequence of integers into a target sequence by means of a series of transpositions where the chunks that are transposed can be reversed in order. Solutions were evaluated according to three criteria: the “Manhattan distance”⁷ between the target sequence and the test sequence after a given number of transpositions; the efficiency of the code in terms of the run time of the program; and the readability of the code in terms of the maximum cyclomatic complexity of its functions.

Like *étappes* in the Tour de France, this contest had several stages. Four of them, to be precise: during the first day of the contest, you couldn't see the code or scores for any of the entries –this was called “Darkness”; during the second day of the contest, you could see scores but no code –“twilight;” for the remainder of the contest you could see scores and code for all entries –“daylight” –with the exception of one day “late-stage twilight” in which you could see scores but no code. Again, like in the Tour de France, this contest had several intermediate winners: winners were determined at the end of each stage and in addition there was an “early bird” winner and a “best result by 5:00 PM winner.”

The evolution of the submissions of solutions to the “Splicing” problem, over the contests duration is displayed in summary form in the Figures shown in the text below. At the top (Figure A) shows the improvement (in the downward movements) of the scores of accepted submissions, aligned in time with a bar graph display (Figure B) of the intensity of activity measured as the hour-by-hour count of the number of submission. The graphs presented here suffice to suggest the complex relationships that may exist between the parameters of the contest regime and the contest dynamics that emerge from the micro-level actions and interactions among the participating programmers: at first, during darkness, most of the participants appear to withhold their submission until well into that stage. Inasmuch as nothing was to be gained by early disclosure, allocating one's time to submit before one's score would be revealed to others, or could be assessed against their performance, would not be a sensible competitive strategy and these observations therefore seem quite straightforward to rationalize.

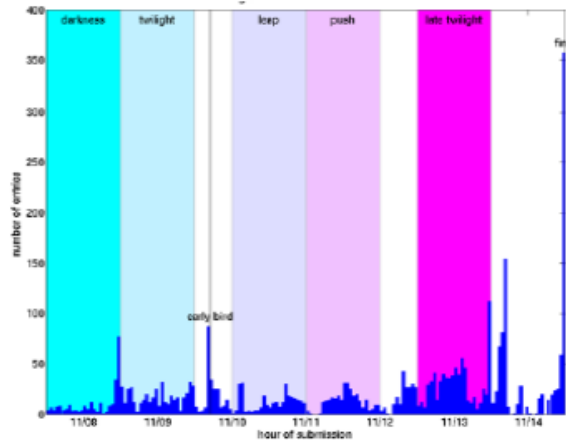
The main difference that appears between the first twilight and the following daylight is in the diversity in solution quality, which is much greater – in normalized proportional terms during the twilight phase. In contrast, the relative diversity among winners is much more pronounced during daylight as many more people can more easily improve upon the previous submissions. The onset of the innovation of an interval of “late twilight,” prior to the return to “daylight” during the closing day of the competition, saw the regeneration of a pronounced degree of diversity in solution quality. It is interesting here is that the advent of twilight did not seem to stem the rate of improvement with respect to the best solutions. The impressive “break-through”, sending the level of best-performance scores sharply downward just after the advent of the “late twilight” does suggest the presence of strategic withholding

⁷ This metric is obtained by computing the distance between two points measured along axes at right angles. In a plane with p_1 at (x_1, y_1) and p_2 at (x_2, y_2) , it is $|x_1 - x_2| + |y_1 - y_2|$. Such quantify matrix resemblance, e.g., by finding row (column) distance as the sum of absolute differences between corresponding elements of two columns (rows).

of enhancements to postpone the opportunity for borrowing by others as the close of the contest approached.

An Experiment with the Disclosure Regime Introducing a "Late Twilight" in the MatLab Competition "Splicing" (Nov. 2007)

Figure A: Evolution of Activity during the "Splicing's" Contest's "Phases"

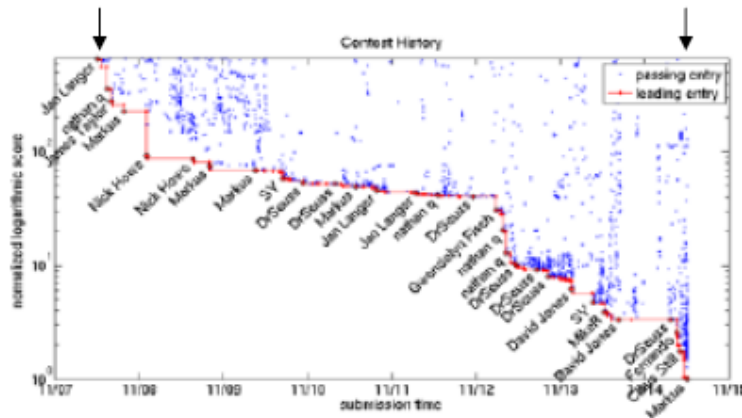


Note: Figure 4b Plots Activity by Hour During the Competition's "Phases"

Each bar represents an hour's worth of entries. The contest has three major phases. The first day is in "darkness", where contestants can submit entries but they can't see any of the entries or their scores. To win this phase, an entry must be general and robust. The second day is "twilight", where we show the scores but not yet the code. This allows contestants to develop their algorithm without anyone else being able to leverage their ideas.

On the histogram, the darkness and twilight phases are the two boxes on the left. The other boxes and vertical gray lines call out other mid-contest challenges: a "late twilight" for the penultimate day of the competition is shaded in purple on the right.

Figure B: Evolution of Normalized Logarithm of Submissions' Scores



Source: <http://www.mathworks.com/contest/splicing/statistics.html>

But the reasons behind the two spikes in activity that came the end of "late twilight" and, shortly thereafter-- early in the opening of the final contest day when others' code was again accessible, would appear to be rather different. The latter burst could well be the result of small improvements being submitted on code that was found to have moved into the forefront during the interval of "late twilight", whereas the former spike could well have reflected the release of code that participants had written during the (twilight) interval when they were not distracted by having access to what others had submitted. The dramatic rush of

submissions in the closing hour of the competition, and the significant relative improvements of performance that was associated with it again seem to reflect strategic withholding to inhibit borrowing and incremental modifications of submissions that would quite likely have been in the lead had they been released earlier.

This points to the possibility that when the time structure of disclosure regimes is known in advance, at least some contestant's activity levels and strategies with regard to the quality of the code that they submit (and thereby may reveal) in "daylight" might be curtailed until they are able to enter it under the cover of twilight. Of course, an alternative possibility remains, namely that simply the disclosure that some individuals have achieved a breakthrough to a notably improved level of performance triggers greater activity among many others and this yields further independent advances by the ensemble. More opportunities to observe the effects of "late twilight" intervals, through experimentation in future contests, and joint analysis of the micro-level association between heightened activity and greater relative rates of advance in performance, will be needed to sort out the strategies and interaction effects among the contestants that underlie these macro-level patterns.

Clearly, the most important point that emerges from the contemplation of Figure 4 is that speculations of the foregoing kind are not the end of analysis, but are simply curiosity-provoking preliminaries to a systematic inquiry into the detailed dynamics of this competition that is made feasible by the available data archive.

References

- [Astrachan, 2004] Astrachan, O. (2004). Non-competitive programming contest problems as the basis for just-in-time teaching. In *Frontiers in Education, 2004. FIE 2004. 34th Annual*, pages T3H/20–T3H/24 Vol. 1.
- [Astrachan et al., 2009] Astrachan, O., Hambruch, S., Peckham, J., and Settle, A. (2009). The present and future of computational thinking. In *Proceedings of the 40th ACM technical symposium on Computer science education*, pages 549–550, Chattanooga, TN, USA. ACM.
- [Axelrod and Hamilton, 1981] Axelrod, R. and Hamilton, W. (1981). The evolution of cooperation. *Science*, 211(4489):1390–1396.
- [Bennett and Lanning, 2007] Bennett, J. and Lanning, S. (2007). The netflix prize. In *Proceedings of KDD Cup and Workshop*, San Jose, CA.
- [Boersen and Phillipps, 2006] Boersen, R. and Phillipps, M. (2006). Programming contests: Two innovative models from new zealand. In *Proceedings of workshop on Perspectives on Computer Science Competitions for (High School) Students*, Dagstuhl, Germany. IBFI Schloss Dagstuhl.
- [Boldrin and Levine, 2008] Boldrin, M. and Levine, D. K. (2008). Perfectly competitive innovation. *Journal of Monetary Economics*, 55(3):435–453.
- [Boudreau and Hagiu, 2008] Boudreau, K. and Hagiu, A. (2008). Platform rules: Multi-Sided platforms as regulators. *SSRN eLibrary*.
- [Boudreau et al., 2008] Boudreau, K., Lacetera, N., and Lakhani, K. R. (2008). Parallel search, incentives and problem type: Revisiting the competition and innovation link. Technology and Operations Management Unit Research Paper 124038, Harvard Business School.
- [Brandt and Messeter, 2004] Brandt, E. and Messeter, J. (2004). Facilitating collaboration through design games. In *Proceedings of the eighth conference on Participatory design: Artful integration: interweaving media, materials and practices - Volume 1*, pages 121–131, Toronto, Ontario, Canada. ACM.
- [Burroughs and Mick, 2004] Burroughs, J. E. and Mick, D. G. (2004). Exploring antecedents and consequences of consumer creativity in a problem-solving context. *Journal of Consumer Research*, 31.
- [Che and Gale, 2003] Che, Y. and Gale, I. (2003). Optimal design of research contests. *American Economic Review*, 93(3):646–671.
- [Cormack et al., 2006] Cormack, G., Kemkes, G., Munro, I., and Vasiga, T. (2006). Structure, scoring and purpose of computing competition. In *Proceedings of workshop on Perspectives on Computer Science Competitions for (High School) Students*, Dagstuhl, Germany.
- [Dagiene, 2006] Dagiene, V. (2006). Information technology contests: introduction to computer science in an attractive way. *Informatics in education*, 5(1):37–46.
- [Dasgupta and David, 1994] Dasgupta, P. S. and P. A. David (1994). Toward A New Economics of Science, *Research Policy*, vol. 23, 1994, pp. 487-521.
- [David, 2002] David, P. A. (2002). Cooperation, Creativity and Closure in Scientific Research Networks: Modelling the Simpler Dynamics of Epistemic Communities,” *Institutions et innovation: de la recherche aux systèmes sociaux d’innovation*, Ed., Jean-Phillipe Touffut, Paris: Bibliothèque Albin Michel (Économie). See also revised version (September 2002) presented to the SIEPR-CREEG Seminar in Technology & Social Science (12/04/2002) [Available at: http://siepr.stanford.edu/programs/SST_Seminars/David_All.pdf.]
- [David, 2007] David, P. A. (2007). Distributed Problem-Solving Organizations and the Evolution of Open Innovation Strategies, Presentation to the Academy of Management Conference: Session on 'Open Source Corporate Strategies', held 6-8 September 2007 (Philadelphia, PA). [Revised version, 10 December 2007, available at: http://www-siepr.stanford.edu/papers/pdf/AOM_OSCS-symposium_pad_v4.pdf .
- [David, 2008] David, P. A. (2008). "The Historical Origins of Open Science: An essay on Patronage, Reputation and Common Agency Contracting in the Scientific Revoulution", *Capitalism and Society* 3(2), November, see <http://www.bepress.com/cas/vol3/iss2/art5> for download of the full text, article; <http://www.bepress.com/cas/announce/20081103> for the whole issue containing a “Discussion” of this article by K. J. Arrow. .
- [Drake et al., 1999] Drake, A. R., Haka, S. F., and Ravenscroft, S. P. (1999). Cost system and incentive structure effects on innovation, efficiency and profitability in teams. *The Accounting Review*, 74(3):323–345. ArticleType: primary_article / Full publication date: Jul., 1999 / Copyright © 1999 American Accounting Association.
- [Eickelpasch and Fritsch, 2005] Eickelpasch, A. and Fritsch, M. (2005). Contests for cooperation–

- A new approach in German innovation policy. *Research Policy*, 34(8):1269–1282.
- [François et al.,] François, J., Kengue, D., Euzenat, J., and Valtchev, P. OLA in the OAEI 2007 evaluation contest.
- [Gulley, 2001] Gulley, N. (2001). Patterns of innovation: a web-based MATLAB programming contest. In *CHI '01 extended abstracts on Human factors in computing systems*, pages 337–338, Seattle, Washington. ACM.
- [Gulley, 2004] Gulley, N. (2004). In praise of tweaking: A wiki-like programming contest. *ACM Interactions*, 11:18–23.
- [Gulley and Lakhani, 2009] Gulley, N. and Lakhani, K. R. (2009). The patterns of innovation generation in a collaborative community: Exploring the relationship between knowledge novelty and reuse. (Presentation by K.R.Lakhani at Beckman Center, Harvard Law School: n.d.).
- [Haller et al., 2009] Haller, J., Bullinger, A. C., and Neyer, A.-K. (2009). Beyond the black box of idea contests. In *Proceedings of the 2009 EURAM Conference on Renaissance and Renewal in Management Studies; Track 14: Innovation - continuing the journey*, Liverpool, UK.
- [Hargadon and Sutton, 1997] Hargadon, A. and Sutton, R. I. (1997). Technology brokering and innovation in a product development firm. *Administrative Science Quarterly*, 42(4):716–749. ArticleType: primary_article / Full publication date: Dec., 1997 / Copyright © 1997 Johnson Graduate School of Management, Cornell University.
- [Hatchuel, 2001] Hatchuel, A. (2001). Towards design theory and expandable rationality: The unfinished program of Herbert Simon. *Journal of Management and Governance*, 5(3):260–273.
- [Hatchuel and Weil, 2009] Hatchuel, A. and Weil, B. (2009). C-K design theory: an advanced formulation. *Research in Engineering Design*, 19(4):181–192.
- [Hooker, 1996] Hooker, J. N. (1996). Testing heuristics: We have it all wrong. *Journal of Heuristics*, 1(1):33–42.
- [III and Noel, 2009] III, G. P. W. and Noel, T. W. (2009). The impact of knowledge resources on new venture performance. *Journal of Small Business Management*, 47(1):1–22.
- [Kenderov, 2008] Kenderov, P. S. (2008), "World Federation of Mathematics Competitions: A Short History". Available at: <http://www.amt.canberra.edu.au/wfnmchistory.pdf>.
- [Kerins et al., 2009] Kerins, B., Simoneau, M., Ezernieks, K., Gulley, N., Thomas, M., Poh, M., and Cetto, L. (2009). The MATLAB programming contest. [<http://www.mathworks.com/contest/overview.html>].
- [Kitano et al., 1997] Kitano, H., Asada, M., Kuniyoshi, Y., Noda, I., Osawa, E., and Matsubara, H. (1997). RoboCup: a challenge problem for AI. *AI Magazine*, 18(1).
- [Kollock, 2003] Kollock, P. (2003). Social dilemmas: The anatomy of cooperation. *Annual Review of Sociology*, 24:183–214.
- [Kozinets et al., 2008] Kozinets, R. V., Hemetsberger, A., and Schau, H. J. (2008). The wisdom of consumer crowds: Collective innovation in the age of networked marketing. *Journal of Macromarketing*, 28(4):339–354.
- [Ladd and Harcourt, 2005] Ladd, B. and Harcourt, E. (2005). Student competitions and bots in an introductory programming course. *J. Comput. Small Coll.*, 20(5):274–284.
- [Lakhani et al., 2007] Lakhani, K. R., Jeppesen, L. B., Lohse, P. A., and Panetta, J. A. (2007). The value of openness in scientific problem solving. Working paper.
- [Moreau and Dahl, 2005] Moreau, C. P. and Dahl, D. W. (2005). Designing the solution: The impact of constraints on consumers' creativity. *Journal of Consumer Research*, 32.
- [Rossi-Doria et al., 2003] Rossi-Doria, O., Sampels, M., Chiarandini, M., Knowles, J., Manfrin, M., Mastrolilli, M., Paquete, L., and Paechter, B. (2003). A comparison of the performance of different metaheuristics on the timetabling problem. In Burke, E. and Causmaecker, P. D., editors, *Practice and Theory of Automated Timetabling IV: 4th International Conference, PATAT 2002, Gent, Belgium, August 21-23, 2002. Selected Revised Papers*, volume 2740 of *Lecture Notes in Computer Science*, pages 329–351, Heidelberg, Germany. Springer-Verlag.
- [Santos et al., 2008] Santos, F. C., Santos, M. D., and Pacheco, J. M. (2008). Social diversity promotes the emergence of cooperation in public goods games. *Nature*, 454(7201):213–216.
- [Sieg et al., 2009] Sieg, J. H., Wallin, M. W., and von Krogh, G. (2009). Managerial challenges in open innovation: a study of innovation intermediation in the chemical industry. In *Proceedings of the 2009 EURAM Conference on Renaissance and Renewal in Management Studies; Track 14: Innovation - continuing the journey*.
- [Terwiesch and Xu, 2008] Terwiesch, C. and Xu, Y. (2008). Innovation contests, open innovation, and multiagent problem solving. *Management Science*, 54(9):1529–1543.

[Tucci and Villarroel, 2007] Tucci, C. L. and J. A. Villarroel, "Open Source Corporate Strategy: Reaching Outside the Boundaries of the Firm to Tap Global Talent," Academy of Management Conference Session Presentation. Philadelphia (6-8 September).

Appendix: Statistical Explorations of Contestants' Activity Levels

A.1 Heterogeneity in activity levels among the population of MatLab contestants

A first approximation of the heterogeneity in activity among participants is given by the distribution of author-ids over all submissions all 8 MatLab competitions, which appears in Figure 1. Clearly, this distribution is highly skewed. Note however that participants can adopt multiple names. Consequently the distribution in terms of participants as opposed to the distribution in terms of the names they have used is likely to be less skewed.

Figure 1: Distribution of submissions per participant over all contests (number of submission is indicated in log scale).

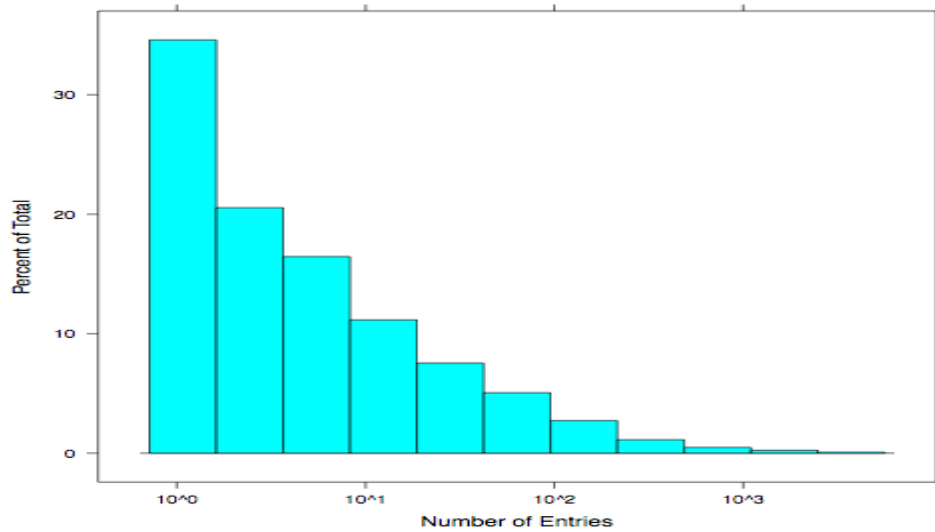
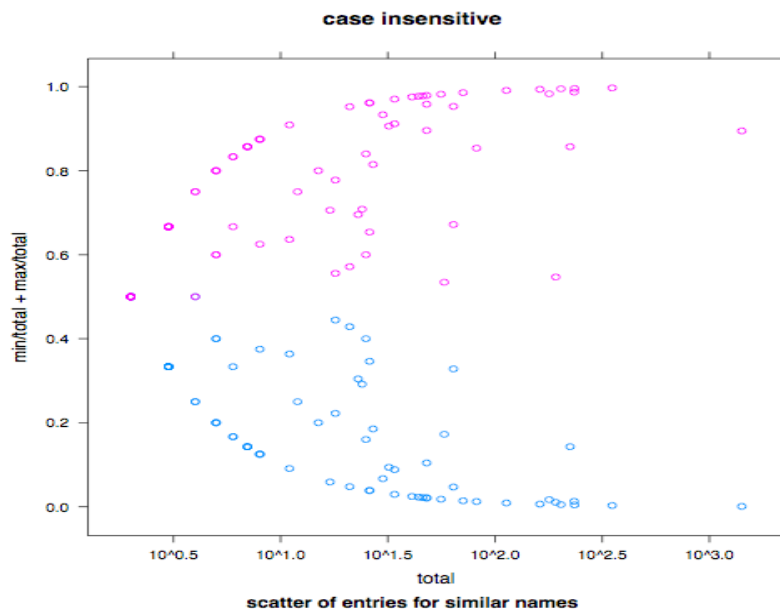


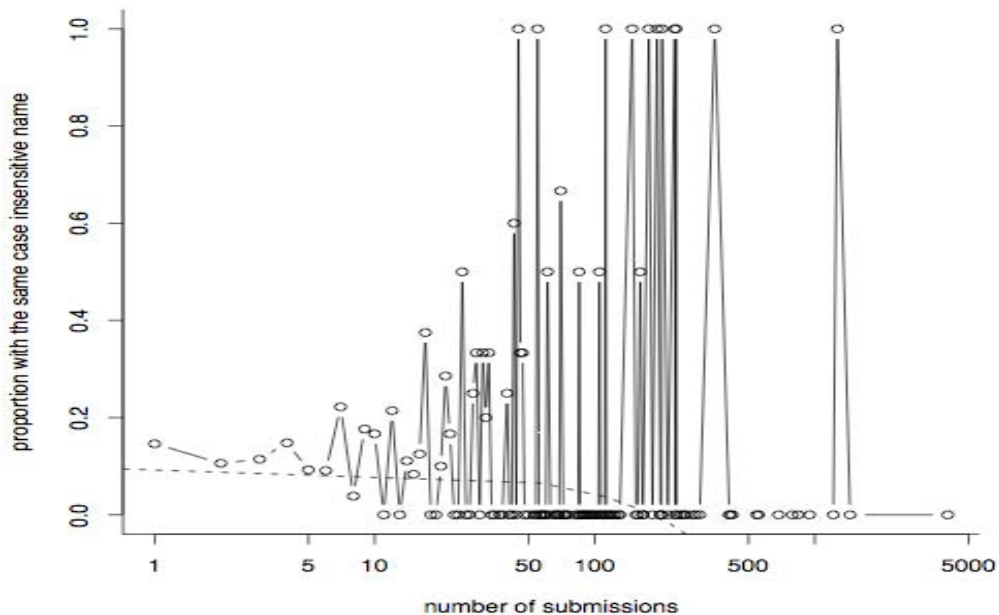
Figure 2: Submissions by groups – mostly pairs – of participants with similar names where similarity is defined as those names that are the same save for letter case differences.



Notes on Figure 2: The dots below $y=0.5$ give the proportion of submissions by the least active name and the dots above give the proportion for the most active name among the group. The x-axis indicates the total number of submissions by the group (in log-scale).

In many instances the use of multiple names by participants is unintended. Often two letters in the name have been swapped, a capitalization-scheme is not consistently adhered to, or white-spaces appear and disappear where they shouldn't have. Typically these mistakes are frequent among the population but rare per individual: For instance, among the 1246 unique names in our database, 167 are the same as another name if we ignore the case of the letter. For vast majority of these names there is only one other variant using a different capitalization. Still there are a few with two other variants and one with three. Figure 2 compares the proportion of total submissions that is attributed to the least frequent name in a pair (in the lower half) to the proportion of submissions associated with the most frequent name (in the upper half). Although there are some exceptions, in most cases a mistaken capitalization only appears once. Moreover, as is illustrated in Figure 3, the propensity to make mistakes of this kind is independent of the number of submissions that a participant makes. Figure 3 shows the proportion of names with variants as a proportion of the total number of names associated with a certain number of submissions. One might have expected some sort of U-shaped curve if the likelihood that a variant of a name appears would be correlated with the number of times that the name is entered. On the contrary, the proportion of case-doubled names seems to be constant at about 10%.

Figure 3 Proportion of names for which the same name with letters in a different case can be found elsewhere, relative to the number of submissions associated with the name. [Dashed line represents the trend (linear model fit)].



A similar analysis can be carried out for other common errors like variations in the amount of white-space used in a name (see Figure 4). The overall conclusion, however, seems to be that problems caused by small variations in the names that participants use can safely be ignored.

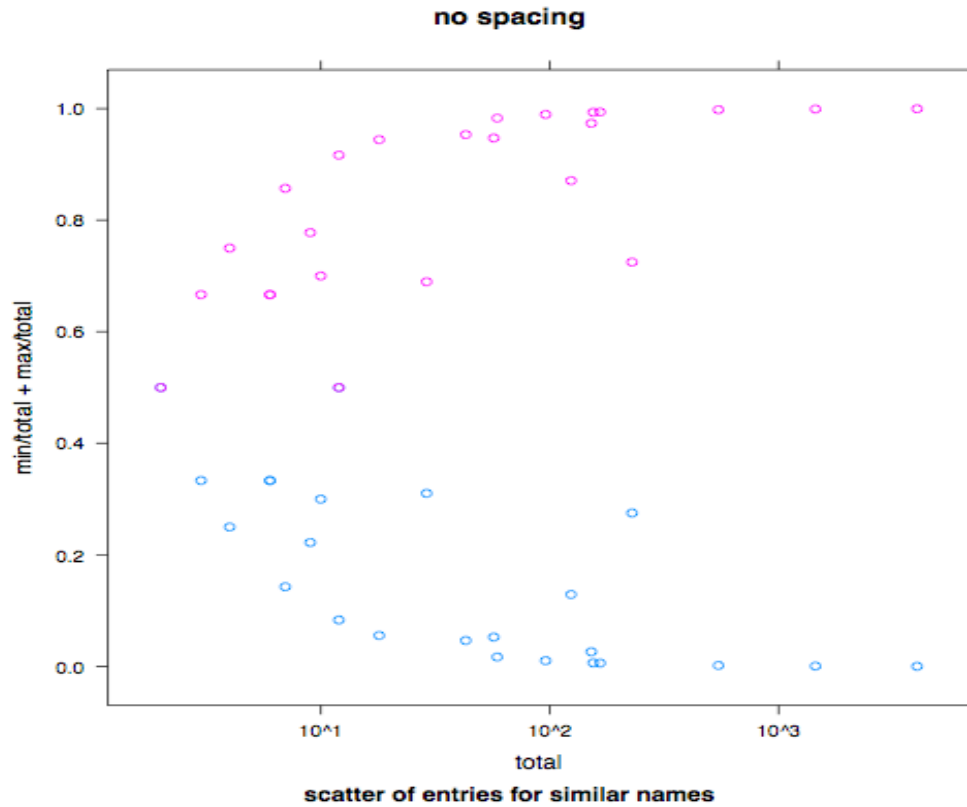


Figure 4. Same analysis as in Figure 2 except that here similarity is obtained by ignoring white-space as well as the characters ‘.’, ‘-’, and ‘_’.

In some respects, however, attributing small variations in a name to a common source does change the overall picture a bit. Table 1 lists the total number of contests in which a name occurs. Clearly, the skew of the distribution decreases the more we gloss over small variations in names. Nevertheless the effect is hardly significant.

Table 1 Number of contests participants have entered

total contests	1	2	3	4	5	6	7	8	9	10	11
author id	1038	120	36	25	11	4	8	1	0	2	1
case insensitive	948	112	41	27	15	4	7	1	1	2	1

Table 2 introduces another restriction: it counts only contests to which participants have submitted at least two times. Also here the differences between the distributions are minor.

Table 2 Number of participants who have entered a given number of contests at least twice.

	0	1	2	3	4	5	6	7	8	9	10	11
author id	446	651	80	28	22	7	3	6	0	0	2	1
case insensitive	381	628	72	34	25	7	3	5	1	0	2	1

Table 3 Number of participants per contest depending on whether capitalization of names is taken into account

	sensitive	insensitive
trucking	121	115
molecule	120	114
protein	178	173
gerrymander	142	138
furniture	75	72
ants	141	137
sudoku	152	148
blockbuster	149	145
blackbox	145	142
jumping	111	107
splicing	119	116

In the foregoing data validation analyses we considered all submissions. Below, we consider submissions that are accepted by the contest server as “eligible”.

Figure 5 and 6 concern the distribution of the length of the period in which people participate in the contest. Figure 5 shows the distribution of length of engagement per participant per contest, and Figure 6 explores the relation between length and intensity of engagement. On average people engage about one day with the contest, but there is quite a bit of variation among participants. The number of candidate solutions that participants submit constitutes a lower bound on the amount of time they engage with the game. Otherwise, there seems to be little relation between both ($\text{cor}(\text{time-span}, \text{submissions})=0.37$).

Figure 5: Boxplot of difference between first and last submission of participants per contest. Only participants with at least two valid submissions are considered.

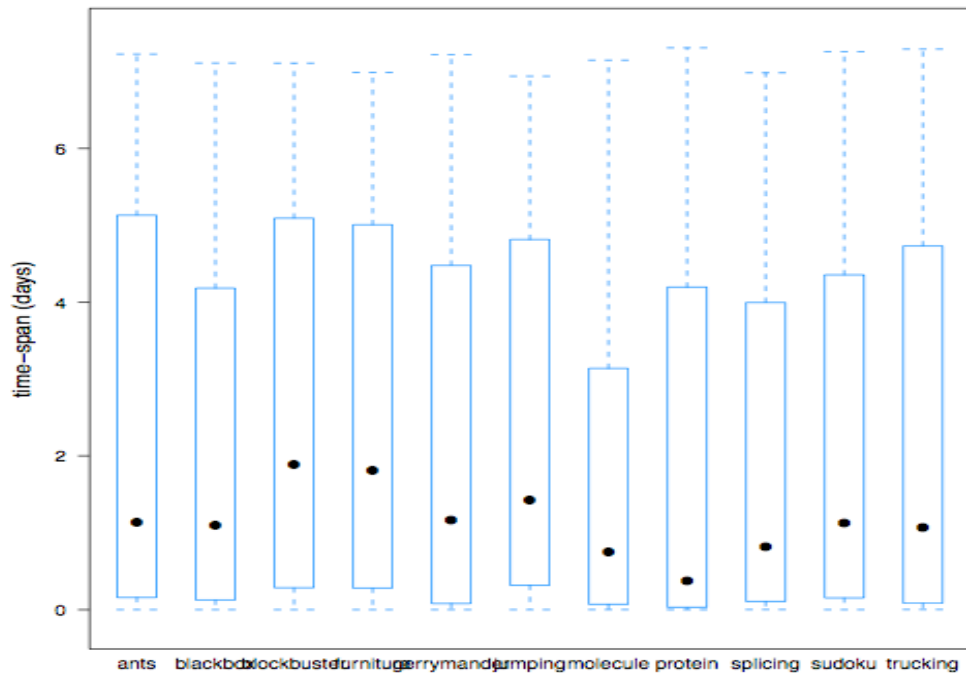
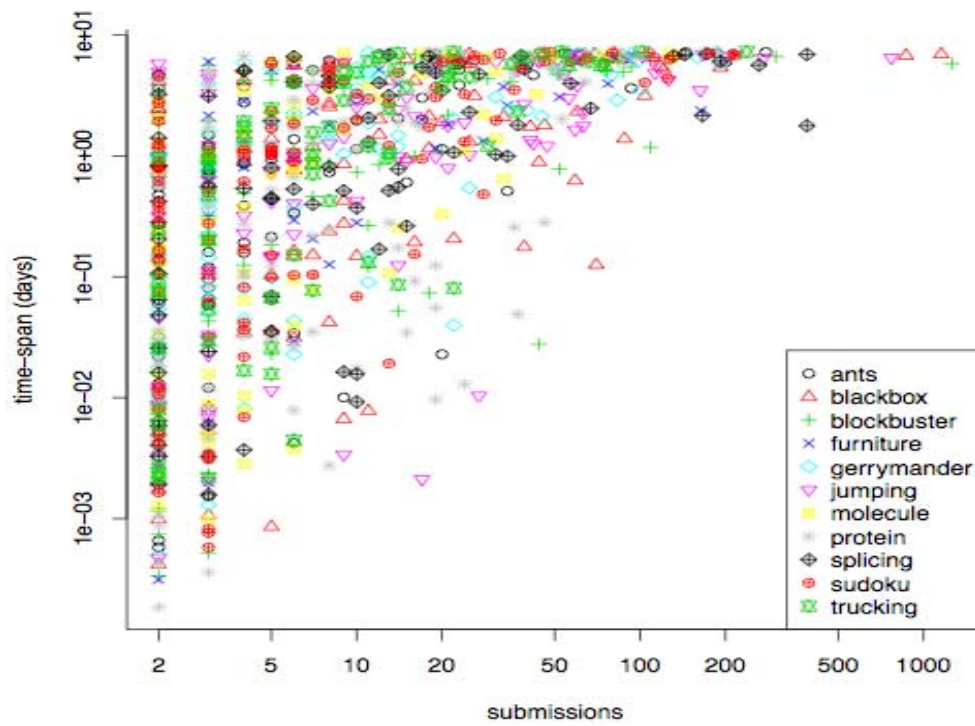


Figure 6 Scatter-plot (log-log scale) of number of submissions against length of engagement for participants in each contest.



A.3 Distribution of MatLab ontestants by relative expertise: Quantile-cohorts:

- The strata of interest are: (i) Ever-leaders; (ii) Ever in top-decile; (iii) Never in top half; (iv) Never it top quarter

Table 4 Number of participants per competition by initial rank of submission

	ants	black box	block buster	furni ture	gerr yma nder	jump ing	mole cule	prot ein	splici ng	sudo ku	truck ing
parti cipa nts	141	145	149	75	142	111	120	178	119	152	121
ever- lead er	18	8	40	28	47	25	23	37	22	31	50
ever- seco nd	13	5	26	17	28	16	7	18	17	22	28

Figure 7: Cumulative distribution of submissions over time per contest

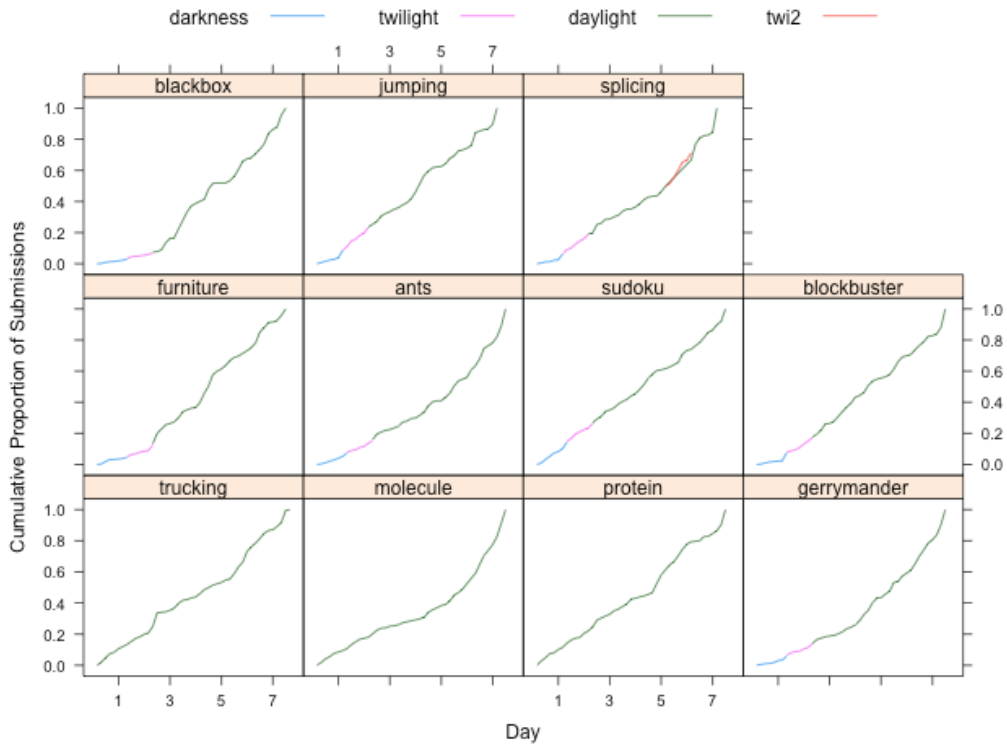


Table 5 Kolmogorov-Smirnov test on submissions per 4-hour period in different regimes

	daylight	twilight	twilight	darkness	daylight	darkness
	D	p	D	p	D	p
gerryman der	0.5133929	0.096842643	0.6071429	0.127518579	0.625	1.35E-02
furniture ants	0.6071429	0.028975979	0.625	0.108210383	0.6875	4.72E-03
sudoku	0.46875	0.160188182	0.3035714	0.881567997	0.5625	3.48E-02
blockbust er	0.3080357	0.647001656	0.3392857	0.783347385	0.25	8.19E-01
blackbox jumping	0.5625	0.052787197	0.6071429	0.127518579	0.78125	8.09E-04
splicing	0.7741935	0.002128466	0.375	0.670041625	0.8387097	2.60E-04
all	0.2165899	0.951576083	0.7142857	0.053030303	0.6313364	2.11E-02
	0.2527473	0.872692475	0.7142857	0.05623007	0.5494505	7.16E-02
	0.2644009	0.003365353	0.3617512	9.04742E-04	0.5362903	8.13E-13

A.4 Komolgorov-Smirnov test on frequency distributions with respect to:

- absolute activity per person
- length of participation
- drop-out rate.

Table 6 Two-sample Kolmogorov-Smirnov test on average activity levels per contestant (alternative=less)

	daylight	twilight	twilight	darkness	daylight	darkness
	D ⁻	p	D ⁻	p	D ⁻	p
gerryman der	2.93E-01	2.93E-03	1.34E-01	0.4622162	0.2872602	9.78E-03
furniture ants	2.86E-01	8.47E-02	2.38E-01	0.2808762	0.4259259	2.41E-03
sudoku	2.65E-01	8.97E-03	2.08E-02	0.9807434	0.2027027	8.18E-02
blockbust er	2.23E-01	2.34E-02	5.18E-02	0.828469	0.1959064	4.23E-02
blackbox jumping	2.16E-01	2.45E-02	1.04E-01	0.5381755	0.2665913	7.72E-03
splicing	2.83E-01	7.57E-03	1.55E-01	0.3687763	0.3292683	1.49E-03
Overall	2.06E-01	6.28E-02	1.99E-01	0.1150692	0.2651822	1.30E-02
	1.79E-01	1.44E-01	1.02E-01	0.63881	0.2342995	7.45E-02
	0.1937652	3.88E-09	0.0833541	0.0716469	0.2327916	4.87E-12

Figure 8: Total number of participants to date per contest and total number of ever-leaders to date for the same contests.

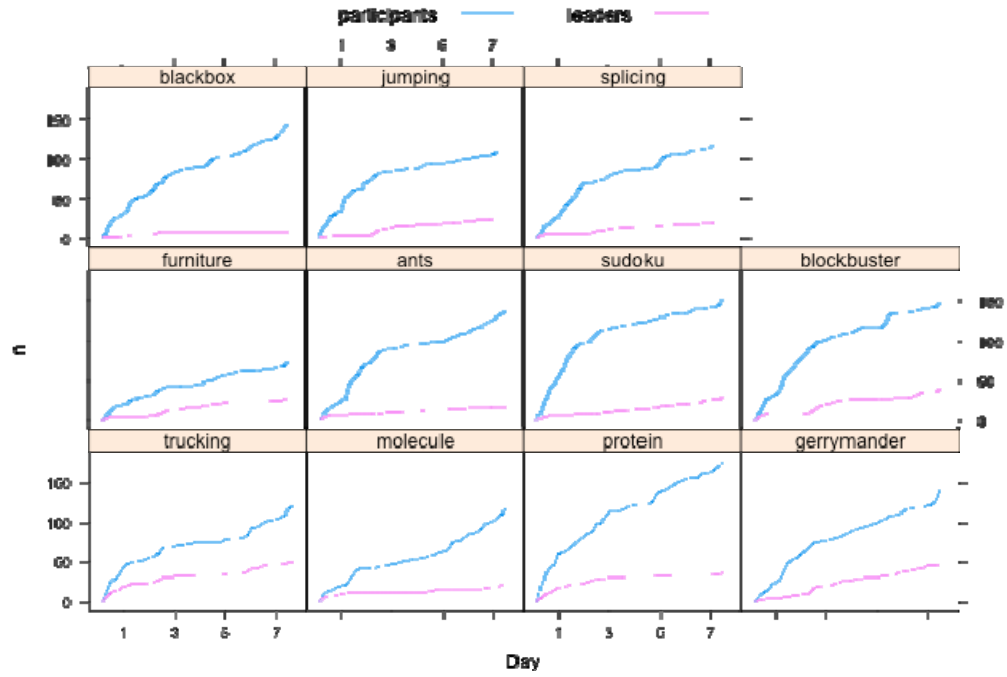


Figure 9 Survivors: number of participants & ever-leaders who appear at least in one later 4-hour period for each 4-hour period in contest.

