

The Implementation of the Turing Tournament: A Report

Jasmina Arifovic *

March 31, 2005

Abstract

This paper provides an overview of the research activities that have already been undertaken regarding the development and implementation of the idea of the Turing Tournament. This is a two-sided Tournament designed to encourage improvement of the existing as well as creation of new models of human behavior, *emulators*, that will be capable of replicating the main features that characterize behavior of experimental human subjects in a variety of economic environments. The other side of the Tournament is represented by the algorithms designed to distinguish between machine and human generated behavior. The paper discusses general design questions and its first implementation within the context of the repeated games. Finally, the paper describes further stages of the Tournament development which will include its implementation in more complicated economic environments with larger strategy space.

1 Introduction

The main goal of social science is to develop good models of human behavior. However, it is not always clear how we know when we have been successful. Econometric methods can tell us which of several models does a better job of

*Department of Economics, Simon Fraser University, Burnaby, BC V5A 1S6, Canada
arifovic@sfu.ca

explaining a given set of data. But the classical econometric way of formulating this question does not really address the question of when the model is “good enough.”

A good example of this problem is in the current literature on learning in repeated games. There are now many competing models to explain how individuals learn in such a setting. Two classical ones are *Fictitious Play* and *Cournot Best Reply*. (See Boylan and El Gamal (1993) for an experimental evaluation of these models). Crawford (1991, 1995) considers *Evolutionary models*. Stahl (1996, 1998) explores boundedly rational rules. Roth and Erev (1998, 1999) use a *Reinforcement Learning* model to explain learning in repeated games. Camerer and Ho (1999a, 1999b) develop the *Experience Weighted Attraction (EWA)* models.

All the above models have been evaluated by using standard econometric methods (usually maximum likelihood methods) to fit the models to experimental data. Using these methods, one can get estimates of the parameters of the model, test various hypotheses about the parameters of the models, and compare models to each other. However, there is no really good way of telling whether the model has done a “good enough” job of representing the underlying decision making process.

Arifovic and McKelvey (2003) explore an alternative approach to evaluating when models of human behavior are “good enough.” They note that if the goal is to mimic human behavior, then the appropriate test to tell if we have done a good enough job is the *Turing Test*. In a famous paper in 1950, Alan Turing addressed the question of determining when computers can “think.” His proposal was to replace this question with the more manageable question of when a computer can mimic human behavior. Turing’s answer was the so called *Turing Test*: a machine is sufficiently human when a third party can not distinguish between the behavior of the machine and a human. In Turing’s version, the third party is a human interrogator who is allowed to ask whatever questions he or she wants to both a machine and a human. Both the machines and humans have their answers put on tape for the interrogator to read.

Arifovic and McKelvey modify the original idea of the Turing test by substituting a computer algorithm for a human interrogator. Thus, their Tournament consists of computer algorithms that they call the *emulators* which can mimic human behavior and of computer algorithms that they call *detectors*, designed to detect whether the observed behavior is generated by the humans or by the emulators. Once all of the entries to the Tournament

are submitted, the first stage involves generation of data, some of which is based on human behavior and some of which is based on the machine behavior. Then the data are presented to the detectors that try to determine which data is human and which is machine generated. The winning detector is the one that does the best job of distinguishing between the human and machine data. The winning emulator is the one that does the best job of fooling the best detector. Thus, unlike the original Turing test that represents an open ended interrogation, their Turing Tournament has the interrogator (detector) and the model of human behavior (emulator) be represented by computer algorithms.

This report proceeds by describing, in section 2, the Turing Tournament in greater detail. An overview of the results of the Initial implementation of the Tournament is given section 3. Finally, section 4 discusses possibilities for future applications of the Turing Tournament methodology to a variety of interesting economic environments.

2 The Turing Tournament - A Description

In the Tournament, the emulators that are submitted generate data sets with information on actions of computer agents in a given environment. The human behavior is represented by datasets generated in the experiments with human subjects in the same environment. The detectors are then presented with all the data sets, both those generated by emulators and by experiments with human subjects, and they try to distinguish between machine and human datasets. They do so by assigning a probability that a given data set is human rather than machine generated. Each detector gets a score based on how close its decisions are to the true state. The detector that obtains the highest score is the winner of the Tournament. The winning emulator is a computer algorithm to which the best detector assigns the highest probability of being human.

It is important to note that the score for a detector is determined by a *proper scoring rule* (the logarithmic proper scoring rule.) This gives incentives for each detector to give a truthful assessment of the posterior probability that each dataset is human. Thus, the winning detector will be the detector whose truthful posterior beliefs are the overall best (given the set of datasets that are presented.) Also, since the winning emulator is the one that does the best job of fooling the best detector, this gives incentives for the

emulators to look as human as possible to the best detector, again providing incentives for developing the best theories of the social behavior in question.

Note that a detector is an algorithm that can contain a variety of different methods for evaluating data including various statistical tests, econometric techniques, data mining methods, algorithmic procedures capable of exploiting some of the well known differences between human and machine generated behavior, etc. In addition, researchers working in the area of learning and experimental economics are well aware of some of the differences, and are able to distinguish between the charts that contain time series of human and those that contain time series of machine generated behavior in various economic settings. Construction of a good detector should actually lead towards formalization and algorithmic expression of the knowledge and intuition that is used when distinguishing between charts that represent human and those that represent machine generated behavior.

The actual development of the Turing Tournament involved addressing a number of design questions. The participants were required to submit their source code in addition to their executable programs. The programs had to be written in such a way to be able to take certain input files supplied by the main Tournament program, and to generate their output in a specified format. (There was a set of requirements relevant for emulators, and the other set relevant for detectors.)

In order for the Tournament methodology to work, it is essential that the incentives be such that the emulators that represent the best models of human behavior in the given setting and the detectors best at distinguishing between human and machine behavior are attracted. In order to guarantee this, it is important that there not be collusion between various participants in the Tournament (three groups of participants, the emulators, the detectors and the human subjects). Thus, one of the Tournament rules specifies that any attempt at collusion is explicit grounds for disqualification. The availability of programs' source code makes any kind of collusion identifiable.

3 Initial Implementation

In order to test the Tournament software that was developed, an internal Tournament was conducted at the California Institute of Technology. The full description of this implementation can be found in Arifovic, McKelvey, and Pevnitskaya (2003). They 'submitted' to the Tournament the source

code for the programs of several emulators (a number of well-known learning algorithms that have been extensively studied in the literature). The main algorithms that were simulated included Fictitious Play, Cournot Best Reply, Adjusted Reinforcement, and Experience Weighted Attractions. In addition, several variants of mixed models where players were using different emulators to make their decisions were submitted. This implementation of emulators and detectors was for illustrative purposes only. While they tried to implement a number of well-known learning algorithms as emulators, and used the parameter sets reported in the literature, Arifovic, McKelvey and Pevnitskaya did not try to compute the ‘optimal’ parameters for each of emulators in each of the games.

The programs for several relatively simple detectors were also submitted. These detectors compute various measures using presented datasets, such as closeness to Nash equilibrium, closeness to payoff dominant outcome, changes in players’ payoffs overtime etc. Based on the values of these measures, detectors give a probability that a particular dataset is human. These detectors represented just an initial attempt to tackle the problem of developing this type of algorithms. They were based on some of the well known differences between human and machine generated data.¹

For this ‘test’ Tournament, experimental data collected by McKelvey and Palfrey (2001) were used. The games that were considered were: Ochs Game, Stag Hunt, Ultimatum game, Centipede game, Prisoner’s Dilemma, Battle of Sexes and the game of Chicken. Machine datasets were generated for the above games using the programs developed for various learning algorithms.

These initial simulations showed significant differences between human and computer generated data. To illustrate the differences, we mention couple of points in this report (the rest of the discussion can be found in Arifovic et al. (2003)). The first point to notice shows up in the data for the Battle of the Sexes game. None of the learning models were able to mimic the coordination that occurs in human data. In the human data, subjects would frequently achieve more than could be achieved by independent randomization by alternating back and forth between the pure strategy equilibria. Thus, on odd moves, they would go to the equilibrium preferred by one of the players, and on even moves to the equilibrium preferred by the other. The emulators did not match this. As a result, the ”coordination detector”

¹A concept of detectors is a new one and a real challenge in the Tournament is to develop good performing detectors.

that tried to detect intertemporal coordination in the Battle of Sexes game was very successful in distinguishing between human generated and computer generated data.

Another observation can be made from the data for the Prisoner's Dilemma game. None of the existing models of learning that were implemented achieved as much cooperation as human subjects did and as a result the average individual payoffs observed in the experiments are much higher than those obtained by the emulators.

The results showed there was room for improvement in developing new emulators or more appropriate and better implemented versions of the existing emulators. The emulators that were implemented were really not designed to take into account either the repeated character of the game, or the fact that the opponent was also learning over time. The differences between human and computer behavior demonstrate that there is room for development of good detectors as well. Building good detectors represents development of a new methodology for evaluating models of human behavior. Building a detector requires one to really think about how humans behave. In addition, better detectors will force improvements in the emulators, the models of human behavior.

4 Implementation during summer 2003

These preliminary results, reported in Arifovic, McKelvey, and Pevnitskaya, served as motivation to conduct a full-scale public Tournament by inviting submissions of better adjusted and more sophisticated emulators and detectors. The Turing software that runs iteratively until, in statistical terms, a significantly winning detector and emulator are identified was developed at the California Institute of Technology and was ready to be implemented in a real Tournament. The organizers of the Tournament, the Turing Group² announced the beginning of the first official Tournament in March of 2003. The deadline for the emulator and detector submissions was May 31, 2003.

A detailed description of the Tournament, its rules and how it was going to be conducted was made available on the Tournament's web site.³ The Turing group gave a list of games that would be used, and the lower and upper bounds for the payoffs for each of the games from the list. In addition,

²Jasmina Arifovic, John Ledyard, Walter Yuan, and Chris Crabbe

³The Turing Tournament web site is <http://turing.ssel.caltech.edu>.

they conducted a new set of experiments to be used for testing purposes only. The human subjects were California Institute of Technology undergraduate students. The samples of these data were also made available on the web site in order to provide developers of emulators and detectors with the data that can be used for testing purposes.

After the deadline, a new set of experiments with human subjects using the set of games that algorithms were later tested on was conducted. Thus, new human datasets were created. At the same time, the programs were tested in order to check if they could be successfully implemented and used with the Turing Tournament software. Once the testing was over, the Tournament was conducted for 10,000 iterations in order to ensure that the winning detector and the winning emulator have statistically significant scores. The Turing group is now working on finalizing the presentation of the results that will be announced shortly. Both the winning detector, and the winning emulator will get a prize of US \$10,000 each.

When the computational part is over and the winners are announced, the Turing group will start the analysis of the submitted algorithms. The objective is to study what it is about the good emulators that distinguishes them from those that do not perform as well, and what characterizes good detectors.

5 Other Applications

The next stage of this research program involves a new application of the Tournament to a more complicated environments in terms of the strategy space and number of players. We will proceed to an implementation in public good environments. This will raise the technical difficulty of both running the Tournament and creating emulators and detectors that can cope with larger numbers of players (greater than 2) and larger strategy spaces. But it will also reveal whether the Tournament technology can be effectively used on more than very limited set of environments. With obvious modifications, the basic methodology, described above for learning in two person games, will have applications in several areas of study. Examples are:

- Studying how cooperation and coordination develop in repeated normal form games, and how it depends on the information and matching conditions.

- Modeling behavior in public good provision problems.
- Explaining bidding behavior and convergence to equilibrium in experimental economic markets.
- Studying and detection of computerized trading in various asset markets (e.g. stock markets, foreign exchange markets).
- Detection of “program trading” in financial markets, i.e. development of a methodology for distinguishing between human traders and program trading.
- Development of methods to detect “market bots” (auction bots, shop bots) on internet auction sites.
- Design of robot agents for use in laboratory experiments.
- Design of machine translation programs.

The Tournament raises fundamental unsolved problems in game theory, computer science, econometrics/statistics, and experimental economics. The expectations are that it will attract general interdisciplinary interest and attention.

References

- [1] Arifovic J, McKelvey RD (2003) The Turing Tournament: A Method for Evaluation of Social Science Theories. Manuscript
- [2] Arifovic J, McKelvey RD, and Pevnitskaya S (2003) An Initial Implementation of the Turing Tournament to Learning in Two Person Games. Manuscript, available at <http://turing.ssel.caltech.edu/index.html>.
- [3] Boylan R and El Gamal M (1993) Fictitious Play: A Statistical Study of Multiple Economic Experiments. *Games and Economic Behavior* 5:205-222.
- [4] Camerer CF, Ho TH (1999a) Experience-weighted Attraction Learning in Games: Estimates from Weak Link Games. In Budescu D, Erev I, Zwick R (eds) *Games and Human Behavior: Essays in Honor of Amnon Rapoport*. Erlbaum, 31-51.

- [5] Camerer CF, Ho TH (1999b) Experience-Weighted Attraction in Games. *Econometrica* 67:827-874
- [6] Crawford, V (1991) An ‘Evolutionary’ Interpretation of Van Huyck, Battalio, and Beil’s Experimental Results on Coordination. *Games and Economic Behavior* 3:25-59.
- [7] Crawford V (1995) Adaptive Dynamics in Coordination Games. *Econometrica* 63 :103-143.
- [8] Erev I, Roth AE (1998) Predicting How People Play Games: Reinforcement learning in Experimental Games with Unique, Mixed Strategy Equilibria. *American Economic Review* 88:848-881.
- [9] Erev I, Roth AE (1999) On the Role of Reinforcement Learning in Experimental Games: The Cognitive Game Theory Approach. In Budescu D, Erev I, Zwick I (eds) *Games and Human Behavior: Essays in Honor of Amnon Rapoport*, Erlbaum, 53-77
- [10] McKelvey RD, Palfrey TR (2001) Playing in the Dark: Information, Learning, and Coordination in Repeated Games. Manuscript.
- [11] Roth AE, Erev I (1995) Learning in Extensive-Form Games: Experimental Data and Simple Dynamic Model in the Intermediate Term. *Games and Economic Behavior*, Special Issue: Nobel Symposium 8:164-212
- [12] Stahl DO (1998) Evidence Based Rules and Learning in Symmetric Normal Form Games. *International Journal of Game Theory* 28:111-130
- [13] Stahl DO (1996) Boundedly Rational Rule Learning in a Guessing Game. *Games and Economic Behavior* 16:303-330.
- [14] Turing A (1950) Computing Machinery and Intelligence. *Mind* 59:433-460.