Probabilistic Approaches to Cheating Detection in Online Games

Laetitia Chapel, Dmitri Botvich and David Malone

Abstract— Cheating is a key issue in online games, as it reduces the satisfaction of honest players with a game and can result in reduced revenue for game providers. Cheat prevention techniques used today typically aim to identify that a particular known technique for cheating is in use, and then prevent that. In this paper we present two techniques for identifying cheating based purely on the results of games. These techniques are based on the law of large numbers and the Bradley-Terry model, and aim to identify statistical behaviour that suggests a player is cheating. Preliminary simulation studies suggest that such techniques are a promising way to identify cheaters.

I. INTRODUCTION

Online games are more and more popular. In particular, Massively multiplayers online games (MMOGs) are video games played on the Internet which involve simultaneously a large number of players. While the popularity of MMOGs has increased dramatically in the last years [1], [2], with some games like *World of Warcraft* with a subscriber base of more that 11 million users[1] and revenues of millions of dollars for gaming companies, cheating has emerged as a notable phenomenon, with its own economy [3] and even companies offering cheating products [4].

Cheating in online games, defined as the set of "*activities that modify the game experience to give one player an advantage over another player(s)*" [4], is a major concern for the gaming companies as it degrades the experience of the honest players and decreases their revenue [5], [6]. They thus try to introduce mitigation schemes in order to prevent cheating or detect the cheaters. The anti-cheating protocols are based on the prevention of known real-world cheats (for a classification of them, see [7]) but it is well-known that they usually introduce complications into the games and some recent works suggest that strategies based on cheat detection should be preferred ([8] and references therein). Authors have also proposed looking for anomalous in-game behaviour to identify cheating [9], through real-time monitoring of gamers.

In this paper, we present a framework for modeling games and cheaters that can be used for cheat detection. The approach can be used in different types of online gaming, from MMOGs to First-Shooter-Games. Probably the most important assumption we make here is related to player ranking. We assume that each player has a rank $\pi_i \geq 0$

Laetitia Chapel is with the Hamilton Institute, National University of Ireland Maynooth (corresponding author, email: `Laetitia.Chapel@nuim.ie`).

Dmitri Botvich is with TSSG, Waterford IT Ireland (email: `dbotvich@tssg.org`).

David Malone is with the Hamilton Institute, National University of Ireland Maynooth (email: `David.Malone@nuim.ie`).

[1]See the press release of Blizzard company `http://us.blizzard.com/en-us/company/press/pressreleases.html?081121`

which determines the probability of the outcomes of their games. We assume that while cheating, the player will have an artificially inflated rank, and we attempt to detect this. Note that a player's rank π_i does not correspond to any in-game ranking, but can be estimated from game results. We use the Bradley-Terry model [10], [11] to estimate the ranks of the players as a function of their performances against the other players. We then construct a probabilistic model for the result of the games between different players and use statistical tests to detect potential cheaters.

To provide evidence that statistical methods can be used to efficiently detect cheaters, we present in this paper two different detection methods.

The first framework that we consider is an extension of the Bradley-Terry model. The Bradley-Terry model provides us with a useful tool for estimating ranks. We define a cheater as a player with the ability to increase artificially their rank (by any cheats) in order to improve their results. We introduce an additional parameter on the model, called the cheating strength, and consider that one player is a cheater when their cheating strength is significantly greater than expected.

Another way of identifying a cheater is to look for players whose mean results are statistically unlikely with respect to their rank. This second framework is based on the limit theorems from probability theory including the central limit theorem. For simplicity we will refer to it as the law of large numbers based algorithms. We note that the second framework makes the additional assumption that the ranks are known in advance or that they slowly evolve with time, and so can be well estimated.

We bring together these two different frameworks as they compliment each other. In particular, the initial estimation of ranks can be performed by the Bradley-Terry based algorithm and then the estimated ranks can be used by the law of large numbers based algorithm for cheat detection.

The remainder of this paper is organized as follows: we first introduce a probabilistic game modeling in section 2. In section 3 we propose two models of cheating and cheat detection algorithms. Section 4 presents some experiments and evaluates the performance of the algorithms. We then conclude and draw some perspectives.

II. PROBABILISTIC GAME MODELING

We suppose that there are $N \geq 2$ players $i = 1, ..., N$ competing in some set of tournaments. Notice that in online games N may be very large ($N \approx 1000$ or even more), but it can also be relatively small. Each player i plays against all the others players $j \neq i$ at least once. Say that X_{ij} is a random variable that represents the number of times player i beats player j, with $X_{ii} = 0$ and $s_{ij} > 0$ is the number of

times player i and j played together.

Players ranking. We assume that each player i has a rank π_i that is normalized such that $0 \leq \pi_i \leq 1$. We note that a player's rank π_i does not correspond directly to any in-game ranking but it can be estimated from game results.

Tournament structure. A tournament is a set of related games between players. We assume that players play in several sequential tournaments with multiple games T_s^m (at least $M = 2$, $m \leq M$ and $s = \sum_{i<j} s_{ij}$). We also consider two types of tournaments: a first one $m = 1$, where the ranks of the players are unknown, and we assume that there are no cheaters during this first round; and all the following tournaments $2 \leq m \leq M$, where the players ranks of the previous tournament $m-1$ are known or have been estimated.

Game outcome. We suppose that the result of a game between players i and j is defined by the so-called outcome function $g(\cdot, \cdot)$, with $0 \leq g(., .) \leq 1$. A player i beats a player j with probability $g(\pi_i, \pi_j) = 1 - g(\pi_j, \pi_i)$. For simplicity, we assume in this paper that $g(\pi_i, \pi_j) = \frac{\pi_i}{\pi_i + \pi_j}$.

III. CHEATING MODELING AND CHEATING DETECTION ALGORITHMS

The crucial element for cheating modeling is the way the cheating behavior is manifested. In order to get insight in cheaters behavior, we first define the cheating strategy and then try to identify cheaters within their underlying strategy. Notice that several strategies can be mixed, which can make the identification of a cheater more complex. Here we will consider only the following ones:

- for the law of large numbers based algorithm, we consider that a cheater cheats systematically. In that case, a cheater i beats non-cheating player j with probability $\hat{g}(\pi_i, \pi_j) > g(\pi_i, \pi_j)$.;
- for the Bradley-Terry framework, we make the assumption that a cheater cheats only with a given number of selected players j. In that case, a cheater i beats player j with probability $\hat{g}(\pi_i, \pi_j) > g(\pi_i, \pi_j)$.

Notice that the assumption that we make for the BT framework can be also used in the LLNs framework, while the opposite does not hold.

A. Law of large numbers based algorithm

The law of large numbers (LLNs) describes the long-term stability of the mean of a random variable. We expect that the results of cheaters will show different long term behavior than non-cheating players. This means that we implicitly assume that cheaters win more than they are supposed to. Note if their cheating strategy makes them win as many games as they would have won without cheating, we won't be able to detect them as a cheater.

Let first consider the case where there are no cheaters in the tournament. In that case, the probability of seeing player i winning k times against player j follows a binomial law $X_{ij} \sim B(s_{ij}, g(\pi_i, \pi_j))$:

$$P(X_{ij} = k) = \frac{s_{ij}!}{k!(s_{ij}-k)!} . g(\pi_i, \pi_j)^k . g(\pi_j, \pi_i)^{s_{ij}-k}$$

and the total number of wins of player i is defined as $X_i = \sum_{j=1}^{N} X_{ij}$. By definition,

$$E(X_i) = E(\sum_{j=1}^{N} X_{ij}) = \sum_{j=1}^{N} E(X_{ij}) = \sum_{j=1}^{N} s_{ij} g(\pi_i, \pi_j)$$

and

$$\sigma_i^2 = V(X_i) = \sum_{j=1}^{N} V(X_{ij}) = \sum_{j=1}^{N} s_{ij} g(\pi_i, \pi_j)(1 - g(\pi_i, \pi_j))$$

as X_{ij} are independent. We note that if there are no cheaters then, by the Central Limit Theorem

$$\frac{X_i - E(X_i)}{\sigma_i} \sim \mathcal{N}(0, 1)$$

for all $i = 1, \ldots, N$ as N approaches infinity. We use here the fact that all game outcomes are independent of each other. We also implicitly assumed that

$$\sup_{i,j} s_{ij} < \infty,$$

i.e. number of games between players are uniformly bounded. Using the normal distribution we can say that without cheaters we get

$$\lim_{N \to \infty} P\left(\frac{X_i - E(X_i)}{\sigma_i} \leq \xi\right) = \Phi(\xi) \qquad (1)$$

where $\Phi(\xi)$ is the cumulative repartition function of $N(0, 1)$. This means that from (1) it follows that with probability $1-\alpha$ the following inequality takes place:

$$X_i \ \leq \ E(X_i) + \xi \cdot \sigma_i.$$

For instance, choosing $\xi = 3$ gives a risk of observing a total number of wins w_i of player i $w_i > E(X_i) + 3 \cdot \sigma_i$ less that 1%.

Now, we consider three other cases, where i is a non-cheating player:

- player j is not a cheater and there are some cheaters in the tournament. We then expect that $X_j < X_i$ as, by assumption, cheaters are players who win more than they are supposed to and they cheat against every other players;
- player k is a cheater and there are no other cheaters in the tournament. In that case, we have $X_k > X_i$ as, by definition, they win more than they are supposed to;
- player l is a cheater and there are other cheaters in the tournament. Then, we can not conclude anything about X_l: if it is a good cheater, we expect that $X_l > X_i$ but if other players are better cheats, we can have $X_l \leq X_i$. In this last case, if the player does not win more than he should have done, he does not fall into the class of cheaters that we are interested in detecting.

All the previous cases hold with probability one (see references on the coupling methods, for instance [12]).

We run N tests, with $n = 1, ..., N$, testing the following hypothesis:

$H_0^n: w_n \ \leq \ E(X_n) + \xi \cdot \sigma_n$

H_1^n: $w_n > E(X_n) + \xi \cdot \sigma_n$
which means that, if we reject H_0^n, we can conclude that player n is not a non-cheater, but if we don't reject H_0^n, we estimate them as a non-cheater, still according to the definition that a cheater is a player who wins more than they are supposed to. So the following algorithm for cheating detection is described:

Algorithm 1 LLNs based cheating detection algorithm

Require: Real ranks π_i, $i = 1, ..., N$. Calculated $E(X_i)$, σ_i and observed w_i
  repeat
    $(i = 1, ..., N)$
    if $(w_i > E(X_i) + \xi \cdot \sigma_i)$ **then**
      player i is detected as a cheater
    else
      player i is detected as a non-cheater
    end if
  until

B. Bradley-Terry extended models

1) Bradley-Terry model for rank estimation: In the first tournament $m = 1$, we make the assumption that they are no cheaters, in order to estimate the real rank of all the players. Otherwise, we can assume that the initial rank of the players are known when they enter a game.

To estimate the rank of each player, we assume that player i beats player j with a probability $g(\pi_i, \pi_j)$. The function $g(\pi_i, \pi_j) = \frac{\pi_i}{\pi_i + \pi_j}$ corresponds to the Bradley-Terry model [10] for paired comparisons. Suppose that the outcomes of all games are independent, and given w_{ij} the number of times player i has beaten player j during the tournament, an approximated rank p_i can be found by minimizing the following negative log-likelihood:

$$\min_{\mathbf{p}} \ell(\mathbf{p}) =$$
$$- \sum_{i<j} \left(w_{ij} \log \frac{p_i}{p_i + p_j} + w_{ji} \log \frac{p_j}{p_i + p_j} \right) \quad (2)$$
$$\text{subject to } 0 \le p_i, i = 1..N, \sum_{i=1}^{N} p_i = 1$$

This problem can be solved using a MM algorithm [13], by the simple iterative procedure Algorithm 2.

Algorithm 2 Rank estimation algorithm

Require: any initial p_j^0, $j = 1, ..., N$, such that $\sum_{j=1}^{N} p_j^0 = 1$
  repeat
    $(k = 0, 1, ...)$
    Let $s = (k \bmod N) + 1$.
    $\mathbf{p}^{k+1} \equiv \left[p_1^k, ..., p_{s-1}^k, \dfrac{\sum_{i,i\neq s} w_{si}}{\sum_{i,i\neq s} \frac{w_{si}+w_{is}}{p_s^k + p_i^k}}, p_{s+1}^k, ..., p_N^k \right]^T$
    Normalize $\mathbf{p}^{k+1}$: $\sum_{i=1}^{N} p_i^{k+1} = 0$
  until $\dfrac{\partial \ell(\mathbf{p}^{k+1})}{\partial p_j} = 0, \forall j$

---

where $\dfrac{\partial \ell(\mathbf{p}^{k+1})}{\partial p_j} = \sum_{i<j} -\dfrac{w_{ij}}{p_i} + \dfrac{w_{ij} + w_{ji}}{p_j + p_i}.$

If $w_i = \sum_{j\neq i} w_{ij} > 0$, $\forall i$, and $s_{ij} > 0$, this algorithm globally converges to the unique minimum of eq. (2). The study of the convergence can be found in [14].

2) Bradley-Terry extended models for cheat detection:

a) Principle: We assume here that we could face cheaters in the tournaments $m > 1$. We propose a model extended from the Bradley-Terry model with home field advantage [15]:

P(player i beats player j) =
$$\begin{cases} \frac{\theta_i \pi_i}{\theta_i \pi_i + \pi_j} & \text{when player } i \text{ has an advantage} \\ \frac{\pi_i}{\pi_i + \theta_j \pi_j} & \text{when player } j \text{ has an advantage} \\ \frac{\pi_i}{\pi_i + \pi_j} & \text{when neither of them has an advantage} \end{cases}$$

with $\theta_i > 0$ the strength of the advantage of player i and π_i their rank at the mth tournament. Having an advantage here means that player i is in a position to cheat, and not the fact that he is actually cheating. Players with associated θ_i significatively different from 1 are the cheaters; when $\theta_i = 1$, $\forall i$, we retrieve the initial Bradley-Terry model.

In this paper, we suppose that, if a player i is a cheater, he is in a position to cheat when its opponent has a better rank. Some other conditions could be used: the only restriction is that we must know on which games they cheat. If we denote π_j^{m-1} the rank of player j on the previous tournament, player i has an advantage on player j (and possibly cheats if he is a cheater) when $\pi_i^{m-1} < \pi_j^{m-1}$, that is to say when he is more likely to lose. We thus obtain the following outcome function:

$$g(\pi_i, \pi_j, \theta_i, \theta_j) = \begin{cases} \frac{\theta_i \pi_i}{\theta_i \pi_i + \pi_j} & \text{if } \pi_i^{m-1} < \pi_j^{m-1} \\ \frac{\pi_i}{\pi_i + \theta_j \pi_j} & \text{if } \pi_i^{m-1} > \pi_j^{m-1} \\ \frac{\pi_i}{\pi_i + \pi_j} & \text{if } \pi_i^{m-1} = \pi_j^{m-1} \end{cases} \quad (3)$$

b) Algorithm: Suppose that the outcomes of all games are independent and that players know the estimated ranks at the last tournament p_i^{m-1} of their opponents, and given w_{ij}^h (or w_{ji}^h) the number of times player i has beaten (or has lost against) player j when $p_i^{m-1} < p_j^{m-1}$ and w_{ij}^a (or w_{ji}^a) the number of times player i has beaten (or has lost against) player j when $p_i^{m-1} > p_j^{m-1}$ and w_{ij}^t (or w_{ji}^t) the number of times player i has beaten (or has lost against) player j when $p_i^{m-1} = p_j^{m-1}$, an approximated rank p_i and strength of the advantage ϑ_i can be found by minimizing the following negative log-likelihood:

$$\min_{\mathbf{p}, \boldsymbol{\vartheta}} \ell(\mathbf{p}, \boldsymbol{\vartheta}) =$$
$$- \sum_{i<j} \left(w_{ij}^h \log \frac{\vartheta_i p_i}{\vartheta_i p_i + p_j} + w_{ij}^a \log \frac{p_i}{p_i + \vartheta_j p_j} + \right.$$
$$w_{ji}^h \log \frac{\vartheta_j p_j}{p_i + \vartheta_j p_j} + w_{ji}^a \log \frac{p_j}{\vartheta_i p_i + p_j} +$$
$$\left. (w_{ij}^t + w_{ji}^t) \log \frac{p_j}{p_i + p_j} \right)$$
$$\text{subject to } 0 \le p_i, i = 1..N, \sum_{i=1}^{N} p_i = 1, \vartheta_i > 0.$$

We look for a simultaneous estimation of $\mathbf{p}$ and $\boldsymbol{\vartheta}$. We use a MM algorithm, taking advantage that we know that, for

positive x and y, $-\ln x \geq 1 - \ln y - \frac{x}{y}$ to construct the surrogate $Q_k(\mathbf{p}, \boldsymbol{\vartheta})$ (see [13] for details):

$$
\begin{aligned}
Q_k(\mathbf{p}, \boldsymbol{\vartheta}) = & -\sum_{i<j} w_{ij}^h \ln \vartheta_i p_i + w_{ij}^a \ln p_i + w_{ij}^t \ln p_i + \\
& w_{ji}^a \ln p_j + w_{ji}^h \ln \vartheta_j p_j + w_{ji}^t \ln p_j + \\
& (w_{ij}^h + w_{ji}^a)\left(1 - \ln(\vartheta_i^k p_i^k + p_j^k) - \frac{\vartheta_i p_i + p_j}{\vartheta_i^k p_i^k + p_j^k}\right) + \\
& (w_{ij}^a + w_{ji}^h)\left(1 - \ln(p_i^k + \vartheta_j^k p_j^k) - \frac{p_i + \vartheta_j p_j}{p_i^k + \vartheta_j^k p_j^k}\right) + \\
& (w_{ij}^t + w_{ji}^t)\left(1 - \ln(p_i^k + p_j^k) - \frac{p_i + p_j}{p_i^k + p_j^k}\right),
\end{aligned}
$$

with $Q_k(\mathbf{p}, \boldsymbol{\vartheta}) \leq \ell(\mathbf{p}, \boldsymbol{\vartheta})$ with equality if and only if $(\mathbf{p}, \boldsymbol{\vartheta}) = (\mathbf{p}^k, \boldsymbol{\vartheta}^k)$. Such a function is said to minorise $\ell(\mathbf{p}, \boldsymbol{\vartheta})$ at the point $(\mathbf{p}^k, \boldsymbol{\vartheta}^k)$. The advantage of using $Q_k(\mathbf{p}, \boldsymbol{\vartheta})$ is that it makes easier the maximization than the original likelihood because it separates the components of $(\mathbf{p}, \boldsymbol{\vartheta})$.

Here, the presence of $\theta_i \pi_i$ leads to a non-separation of the parameters by the surrogate. However, we can construct a cyclic algorithm in that case, first by maximising $Q_k(\mathbf{p}, \boldsymbol{\vartheta}^k)$ as a function of $\mathbf{p}$ and $Q_k(\mathbf{p}^{k+1}, \boldsymbol{\vartheta})$ as a function of $\boldsymbol{\vartheta}$.

Setting $\dfrac{\partial Q_k(\mathbf{p}, \boldsymbol{\vartheta}^k)}{p_i} = 0$ leads to

$$
p_i^{k+1} = w_i \cdot \left[\sum_{j, j \neq i} \frac{g_i^h \vartheta_i^k}{\vartheta_i^k p_i^k + p_j^k} + \frac{g_i^a}{p_i^k + \vartheta_j^k p_j^k} + \frac{g_i^t}{p_i^k + p_j^k}\right]^{-1}
$$

where w_i is the total number of wins of player i, g_i^h the number of games player i played at home, g_i^a away and g_i^t when none of the players has an advantage.

Setting $\dfrac{\partial Q_k(\mathbf{p}^{k+1}, \boldsymbol{\vartheta})}{\partial \vartheta_i} = 0$ leads to

$$
\vartheta_i^{k+1} = g_i^h \cdot \left[\sum_{j, j \neq i} \frac{g_i^h p_i^{k+1}}{\vartheta_i^k p_i^{k+1} + p_j^{k+1}}\right]^{-1}
$$

where g_i^h is the number of wins of player i when they play with an advantage ($p_i^{m-1} < p_j^{m-1}$).

Similarly to the Bradley-Terry model for rank estimation, the convergence of the global algorithm is guaranteed if each player wins and loses at least once away and at home (proof will be presented in a forthcoming paper). Then, with function $g(\pi_i, \pi_j, \theta_i, \theta_j)$ defined in equation (3), these conditions will never be met as the best player a will never play at home (his rank p_a^{m-1} is greater than all the other one) and as the worse player b will never play away (they have smallest rank).

To deal with this problem, we make two additional assumptions: $\theta_a = \theta_b = 1$. For the best player, the assumption makes sense as this player has no opportunities to cheat anyway; for the worse player, the legitimacy of the assumption is less obvious but needed: if the worst player of the previous tournament cheats, we won't be able to detect it.

Algorithm 3 summarises up the simultaneous rank and cheating strength estimation process.

Algorithm 3 Rank and strength estimation algorithm

Require: any initial p_j^0, $j = 1, ..., N$, such that $\sum_{j=1}^{N} p_j^0 = 1$
Require: initial $\vartheta_j^0 = 1$, $\forall j$
  repeat
    $(k = 0, 1, ...)$
    Let $s = (k \bmod N) + 1$.
    $p_s^{k+1} =$

$$
w_s \cdot \left[\sum_{i, i \neq s} \frac{g_s^k \vartheta_s^k}{\vartheta_s^k p_s^k + p_i^k} + \frac{g_s^a}{p_s^k + \vartheta_i^k p_i^k} + \frac{g_s^t}{p_s^k + p_i^k}\right]^{-1}
$$

    $\mathbf{p}^{k+1} \equiv \left[p_1^k, ..., p_{s-1}^k, p_s^{k+1}, p_{s+1}^k, ..., p_N^k\right]^T$
    Normalize $\mathbf{p}^{k+1}$
    if s is not the best or the worst player of the previous tournament **then**
      $\vartheta_s^{k+1} = w_s^h \cdot \left[\sum_{i, i \neq s} \frac{g_s^h p_s^{k+1}}{\vartheta_s^k p_s^{k+1} + p_i^{k+1}}\right]^{-1}$
    else
      $\vartheta_s^{k+1} = \vartheta_s^k$
    end if
    $\boldsymbol{\vartheta}^{k+1} \equiv \left[\vartheta_1^k, ..., \vartheta_{s-1}^k, \vartheta_s^{k+1}, \vartheta_{s+1}^k, ..., \vartheta_N^k\right]^T$
  until $\dfrac{\partial \ell(\mathbf{p}^k, \boldsymbol{\vartheta}^k)}{\partial p_j} = 0$ and $\dfrac{\partial \ell(\mathbf{p}^k, \boldsymbol{\vartheta}^k)}{\partial \vartheta_j} = 0$, $\forall j$

---

c) Tests for detecting cheaters: A cheater is a player whose θ_i value significantly deviates from 1 (or which is significantly greater than 1). We here construct N tests, with $n = 1, ..., N$:
H_0^n: $\boldsymbol{\vartheta} = [\vartheta_1, \vartheta_2, ..., 1, ..., \vartheta_N] = \boldsymbol{\vartheta}_0^n$
H_1^n: $\boldsymbol{\vartheta} = [\vartheta_1, \vartheta_2, ..., \vartheta_n, ..., \vartheta_N] = \boldsymbol{\vartheta}_1^n$, with $\vartheta_n > 1$
If we reject H_0^n, we conclude that player n is a cheater. We proceed by using a likelihood ratio test: we construct the statistics $D^n = 2(\ell(\mathbf{p}, \boldsymbol{\vartheta}_1^n) - \ell(\mathbf{p}, \boldsymbol{\vartheta}_0^n)) \sim \chi^2(N-1)$ where $\ell(\mathbf{p}, \boldsymbol{\vartheta}_1^n)$ is the log-likelihood under the alternative hypothesis and $\ell(\mathbf{p}, \boldsymbol{\vartheta}_0^n)$ the log-likelihood under the null hypothesis. If we reject H_0^n at a significance level α, we conclude that player n is a cheater.

IV. PERFORMANCE EVALUATION ON SIMULATED EXAMPLES

In this section we present simulations to show that our technique can be effective. We first consider the test based on the law of large numbers, which assumes that the ranks are known. Then we will move to a more general situation using the Bradley-Terry extension to track cheaters across multiple games.

A. Data generation

For every tournament, we generate data for $N = 5, 20$ and 80 players, and with $s_{ij} = 5, 10, 20, 40$ and 80 games between each pair of players (except for 5 players and 5 games, where the conditions of the extended Bradley-Terry algorithm convergence are infrequently met).

1) First tournament $m = 1$: Here, we generate N random numbers u_i uniformly distributed on the $[0, 1]$ interval and normalize them such that $\sum_{i=1}^{N} u_i = 1$, and set them as the

true ranks $\pi_i = \pi_i^1 = u_i$. We then simulate a tournament where each pair of players (i,j) plays s_{ij} times against each other. For each game, we generate a random number $u \sim U(0,1)$ and declare player i as the winner if $u < g(\pi_i, \pi_j)$ and as the loser otherwise. In order to obtain a measure of the accuracy of our results, we replicate this tournament 500 times.

2) Next tournament $m = 2$: We first suppose that the true rank of the players evolves between two tournaments $\pi_i^m = \pi_i^{m-1} + u_i$ where $u_i \sim U(0, \left|\frac{1}{N} - 0.2\right|)$ (there can not be a too big gap between the previous and the current rank) and the ranks are $\pi_i = \pi_i^m$ after normalization $\sum_{i=1}^N \pi_i^m = 1$. We consider that we face q cheaters, $0 \leq q \leq N$. For the non-cheating players, we assign $\theta_i = 1$ and for the cheaters $\theta_i > 1$. We test 3 cases: where there are no cheaters, 3 cheaters (with $\theta_i \in \{5, 10, 20\}$) or half of the players are cheaters (with $\theta_i \in \{5, 8, 11, 14, 17\}$).

To determine the result of a game between player i and j, we generate a random number $u \sim U(0,1)$ and declare player i as the winner if $u < g(\pi_i, \pi_j, \theta_i, \theta_j)$ and player j otherwise. In order to obtain a measure of the accuracy of our results, we replicate this tournament 500 times.

B. Results for the LLNs based algorithms

To use the LLNs framework, we have to make the assumption that, either the ranks are known or that they evolve slowly between the tournaments. In the following experiments, we suppose that the ranks are known and use algorithm 1 to discriminate cheaters and non-cheaters, setting $\pi_i = \pi_i^m$.

Table 1 gives the confusion matrix for different combinations of players, games and cheaters, using a test at 95% confidence level. It shows that the algorithm performs well, especially when the number of players and games are high, correctly identifying both cheaters and non-cheaters. When the number of players is small, we observe that the number of cheaters accurately detected as cheaters falls. One possible explanation is that the number of players is too small to satisfy the conditions of the central limit theorem. A second explanation is that, with 3 cheaters and 5 players for instance, at least one of the cheaters will perform less well than the other ones, and then that the cheating strategy doesn't allow one to significantly increase the total number of wins.

The algorithm is fast, performing all the set of experiments in a couple of minutes.

C. Rank estimation on tournament $m = 1$

In the following subsections, we present experiments on π_i and θ_i estimations on simulated data. In implementing Algorithm 2, we use the following stopping criteria:

$$\max_{j,j \in \{1...N\}} \left| \frac{\partial \ell(\mathbf{p}^k)}{\partial p_j} \right| < 0.0001$$

and for Algorithm 3:

$$\max_{j,j \in \{1...N\}} \left| \frac{\partial \ell(\mathbf{p}^k, \boldsymbol{\vartheta}^k)}{\partial p_j}, \frac{\partial \ell(\mathbf{p}^k, \boldsymbol{\vartheta}^k)}{\partial \vartheta_j} \right| < 0.0001$$

which implies that these quantities are close to zero.

We compute the relative mean square error over our 500 simulations in order to evaluate the accuracy of the estimated ranks:

$$\text{RMSE}(\pi) = \frac{1}{500} \sum_{k=1}^{500} \left(\frac{1}{N} \sum_{i=1}^N \frac{(\pi_i - p_i^k)^2}{\pi_i^2} \right)$$

where p_i^k is the estimated rank of player i at the kth replication. Figure 1 shows that the accuracy of the rank estimation increases with the number of players and of games between each pair of players. We see that the estimates are quite good, but improve greatly when the overall number of games becomes larger.
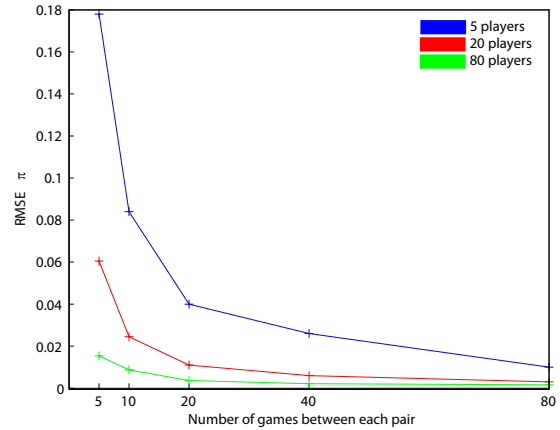

Fig. 1. $RMSE(\pi)$ for the first tournament.

D. Results for the Bradley-Terry model extension

We compute the relative mean square error in order to evaluate the accuracy of the estimated cheating strengths:

$$\text{RMSE}(\boldsymbol{\theta}) = \frac{1}{500} \sum_{k=1}^{500} \left(\frac{1}{N} \sum_{i=1}^N \frac{(\theta_i - \vartheta_i^k)^2}{\theta_i^2} \right)$$

where ϑ_i^k is the estimated strength of player i at the kth replication. Figures 2 and 3 presents the RMSE for the ranks and the strengths of the players. We see that, as for the first tournament, $\text{RMSE}(\pi)$ and $\text{RMSE}(\theta)$ decrease with the number of players and games, and that the number of cheaters seems to have a small influence on the accuracy of the results (except for the $\text{RMSE}(\theta)$ values for 5 players that are different, for which we have no explanation). Nevertheless, we notice that $\text{RMSE}(\pi)$ is larger in the second tournament than in the first, which can be due to the fact that we have in that case more unknown values to estimate.

Table 1 gives the confusion matrix for different combinations of players, games and cheaters. It shows that the tests perform well, identifying correctly the cheaters and the non-cheaters, especially when the number of games are larger. We notice that the number of non-cheaters detected as cheaters is always small, whatever the number of games, players or

TABLE I

C ONFUSION MATRIX FOR 5, 20 AND 80 PLAYERS ON THE TOURNAMENT, WHERE THE PREDICTIONS ARE MADE WITH $\alpha = 0.05$. T HE MATRIX READS AS FOLLOWS: IF WE HAVE 20 PLAYERS PLAYING 5 GAMES, AND IF THERE ARE 3 CHEATERS, THE B RADLEY -T ERM BASED ALGORITHM DETECT 60.3% OF THE CHEATERS (AND THEN DOESN'T DETECT 39.7% OF THEM) WHEREAS IT DETECTS 99.4% OF THE NON-CHEATERS (AND THEN IDENTIFIES AS CHEATERS 0.6% OF THE NON-CHEATERS).

			PREDICTION											
			# cheaters						# non-cheaters					
			0 cheaters		*3 cheaters*		*half cheaters*		*0 cheaters*		*3 cheaters*		*half cheaters*	
			BT	*LLN*	*BT*	*LLN*	*BT*	*LLN*	*BT*	*LLN*	*BT*	*LLN*	*BT*	*LLN*
	# cheater	10 g.			63.7	**60.2**					36.3	**39.8**		
		20 g.			88.3	**70.1**					11.7	**29.9**		
		40 g.			96.7	**76.1**					3.3	**23.9**		
5 players		80 g.			99.7	**83.7**					0.3	**16.3**		
	# non-cheaters	10 g.	3.0	**2.8**	0	**0**			97.0	**97.2**	100	**100**		
		20 g.	3.4	**2.4**	0	**0**			96.6	**97.2**	100	**100**		
		40 g.	2.6	**2.4**	0	**0**			97.4	**97.6**	100	**100**		
		80 g.	3.2	**2.8**	0	**0**			96.8	**97.2**	100	**100**		
	# cheater	5 g.			60.3	**99.9**	61.5	**86.0**			39.7	**0.1**	38.5	**14**
		10 g.			81.0	**100**	81.1	**92.0**			19.0	**0**	18.9	**8.0**
		20 g.			92.0	**100**	91.2	**95.6**			8.0	**0**	8.8	**4.4**
		40 g.			96.7	**100**	97.2	**97.9**			3.3	**0**	2.8	**2.1**
20 players		80 g.			99.7	**100**	99.8	**99.3**			0.3	**0**	0.2	**0.7**
	# non-cheaters	5 g.	0.6	**2.4**	0.6	**0.1**	0.6	**0**	99.4	**97.5**	99.4	**99.9**	99.4	**100**
		10 g.	0.5	**2.5**	0.6	**0**	0.5	**0**	99.6	**97.5**	99.4	**100**	99.5	**100**
		20 g.	0.7	**2.5**	0.6	**0**	8.8	**0**	99.3	**97.5**	99.4	**100**	91.2	**0**
		40 g.	0.7	**2.6**	0.6	**0**	0.9	**0**	99.3	**97.4**	99.4	**100**	99.1	**100**
		80 g.	0.6	**2.3**	0.8	**0**	0.7	**0**	99.4	**97.7**	99.2	**100**	99.3	**100**
	# cheater	5 g.			66.3	**100**	68.1	**95.9**			33.7	**0**	31.9	**4.1**
		10 g.			84.7	**100**	84.0	**98.7**			15.3	**0**	16.0	**1.3**
		20 g.			93.0	**100**	92.1	**99.7**			7.0	**0**	8.0	**0.3**
		40 g.			95.7	**100**	96.1	**100**			4.3	**0**	3.9	**0**
80 players		80 g.			98.7	**100**	98.3	**100**			1.3	**0**	1.7	**0**
	# non-cheaters	5 g.	0	**2.6**	0.3	**0.6**	0	**0**	100	**97.4**	99.7	**99.4**	100	**100**
		10 g.	0.2	**2.6**	0.7	**0.3**	0	**0**	99.8	**97.4**	99.3	**99.4**	100	**100**
		20 g.	0.1	**2.6**	1.0	**0.2**	0	**0**	99.9	**97.4**	99.0	**99.8**	100	**100**
		40 g.	0.1	**2.5**	0.7	**0.1**	0	**0**	99.9	**97.5**	99.3	**99.9**	100	**100**
		80 g.	0.1	**2.5**	1.0	**0**	0	**0**	99.9	**97.5**	99.0	**100**	100	**100**

(ACTUAL — vertical label at left)

cheaters, whereas the number of cheaters detected as non-cheaters is larger when the number of games between players is smaller. This result is not surprising: whereas we have fixed the significance level of the test α (which corresponds to the probability of rejecting the null-hypothesis — we then control the number of false positive), the power of the test $1-\beta$, which correspond to the probability of correctly rejecting the null-hypothesis (here detecting cheaters as cheaters), is not controlled and increased with the sample size.

We also note that, in most cases, this method is less efficient that the LLNs based one. This is expected as the algorithm must estimate all ranks, which the LLN-based scheme assumes are known. The runtime is a couple of hours for the set of experiments, in contrast to a few minutes for the LLN-based algorithms.

V. C ONCLUSION

Cheating is a critical issue in online gaming. We have proposed a framework for modeling games, based on the assumption that the probability of outcomes of the games is only determined by the ranks of the players. Making the (strong) assumption that the ranks are known, we show that a simple law of large numbers based algorithm can be used to detect cheaters. In the case where the ranks are unknown, we use the Bradley-Terry model to estimate them and we derive a new model for cheat detection. We assume that, when cheaters cheat, their rank is artificially increased. We then introduce a new parameter in the Bradley-Terry model, the
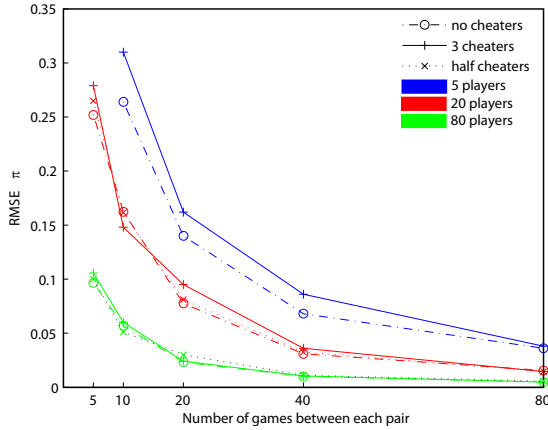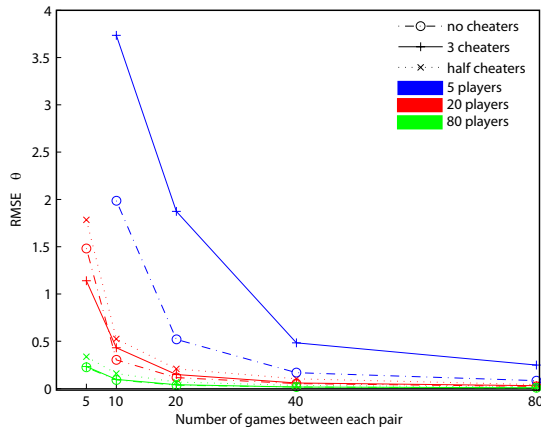
Fig. 2. $RMSE(\pi)$ for the second tournament.

Fig. 3. $RMSE(\theta)$.

strength of cheating, and derive an algorithm that estimates simultaneously the rank and the cheating strength. Experiments on synthetic data show promising results, especially when the number of games is large, and the result accuracy is independent on the number of cheaters.

The use of the LLN based algorithm suppose that the ranks of the players are known or that they are estimated from the previous tournament and evolve very slowly with time. This assumption is valid when we consider games in which the skills of the players do not change quickly, e.g. chess. It would be interesting to study the robustness of the algorithm to the evolution of the ranks. The advantage of using this framework is that the algorithm is fast and no assumptions on the cheating strategy need to be made (no systematic cheating is possible for instance). The use of the Bradley-Terry framework suppose that we know when the players cheat. In this paper, we made the assumption that cheaters cheat only with players better than them but other situations can be considered. The advantage of this framework is that it makes no assumption on the evolution of the skills of the players between two tournaments (as the ranks and cheating strengths are estimated together) and that it quantifies the way players cheat.

Though we get relatively good performance in terms of false positives, we expect that an application of these algorithms would be used to identify players for further investigation. This would allow other factors to be taken into account (say complaints or a reputation system) before the player would be deemed a cheater.

The next step is to evaluate the two cheater detection algorithms on data coming from a real game. We plan to extend this work to consider to consider other types of games or tournaments. We would also like to compare the simpler law of large numbers technique with a rank estimation stage based on a Bradley-Terry model to assess how it behaves when the ranks vary. We are also considering how to use additional information instead of using only ranks and game outcomes. For instance, it could be useful to use the players' opinions about the other players' behavior as a source of input.

VI. ACKNOWLEDGEMENTS

REFERENCES

[1] M. F. Bell, "Toward a definition of "virtual worlds"," *Journal of Virtual Worlds Research*, vol. 1, no. 1, 2008.

[2] R. Schroeder, "Defining virtual worlds and virtual environments," *Journal of Virtual Worlds Research*, vol. 1, no. 1, 2008.

[3] S. De Paoli and A. Kerr, "We will always be one step ahead of them: A case study on the economy of cheating in mmorpgs," *Journal of Virtual Worlds Research*, vol. 2, no. 4, 2010.

[4] Cheating in online games, "Wikipedia, the free encyclopedia," Retrieved february 5, 2010 from http://en.wikipedia.org/wiki/Cheating_in_online_games 2010.

[5] J. Mulligan and B. Patrovsky, *Developping online games: an insider's guide*. New Riders Publishing, 2003.

[6] "Virtual worlds, real money security and privacy in massively-multiplayer online games and social and corporate virtual worlds," European Network and Information Security Agency, Tech. Rep., 2008.

[7] J. Yan and B. Randell, "A systematic classification of cheating in on-line games," in *NetGames'05: Proceedings of the 4th ACM SIGCOMM workshop on Network and system support for games*, 2005, pp. 1–9.

[8] S. Ferretti, "Cheating detection through game time modeling: A better way to avoid time cheats in P2P MOGs?" *Multimedia tools and applications*, vol. 37, no. 3, pp. 339–363, 2008.

[9] P. Laurens, R. F. Paige, P. J. Brooke, and H. Chivers, "A novel approach to the detection of cheating in multiplayer online games," in *ICECCS '07: Proceedings of the 12th IEEE International Conference on Engineering Complex Computer Systems*, 2007, pp. 97–106.

[10] R. A. Bradley and M. E. Terry, "Rank analysis of incomplete block designs. I. the method of paired comparisons," *Biometrika*, vol. 39, pp. 324–345, 1952.

[11] T.-K. Huang, R. C. Weng, and L. Chih-Jen, "Generalized bradley-terry models and multi-class probability estimates," *Journal of Machine Learning Research*, vol. 7, pp. 85–115, 2006.

[12] T. Lindvall, *Lectures on the coupling method*. Wiley, New York, 1992.

[13] K. Lange, D. R. Hunter, and I. Yang, "Optimization transfer using surrogate objectives functions (with discussion)," *Journal of Computational and Graphical Statistics*, vol. 9, pp. 1–59, 2000.

[14] D. Hunter, "MM algorithms for generalized Bradley-Terry models," *The annals of statistics*, vol. 32, no. 1, pp. 384–406, 2004.

[15] A. Agresti, *Categorical data analysis*. Wiley, New York, 1990.