

## **Socializing by Gaming**

### **Revealing Social Relationships in Multiplayer Online Games**

Jia, Adele; Shen, Siqi; van de Bovenkamp, Ruud; Iosup, Alexandru; Kuipers, Fernando; Epema, Dick

**DOI**

[10.1145/2736698](https://doi.org/10.1145/2736698)

**Publication date**

2015

**Document Version**

Accepted author manuscript

**Published in**

ACM Transactions on Knowledge Discovery from Data

**Citation (APA)**

Jia, A., Shen, S., van de Bovenkamp, R., Iosup, A., Kuipers, F., & Epema, D. (2015). Socializing by Gaming: Revealing Social Relationships in Multiplayer Online Games. *ACM Transactions on Knowledge Discovery from Data*, 10(2), 11:1-11:29. [11]. <https://doi.org/10.1145/2736698>

**Important note**

To cite this publication, please use the final published version (if applicable). Please check the document version above.

**Copyright**

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

**Takedown policy**

Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.

## Socializing by Gaming: Revealing Social Relationships in Multiplayer Online Games

Adele Lu Jia<sup>1</sup>, Delft University of Technology  
Siqi Shen, Delft University of Technology  
Ruud van de Bovenkamp, Delft University of Technology  
Alexandru Iosup, Delft University of Technology  
Fernando Kuipers, Delft University of Technology  
Dick H.J. Epema, Delft University of Technology

Multiplayer Online Games (MOGs) like Defense of the Ancients and StarCraft II have attracted hundreds of millions of users who communicate, interact, and socialize with each other through gaming. In MOGs, rich social relationships emerge and can be used to improve gaming services such as match recommendation and game population retention, which are important for the user experience and the commercial value of the companies who run these MOGs. In this work, we focus on understanding social relationships in MOGs. We propose a graph model that is able to capture social relationships of a variety of types and strengths. We apply our model to real-world data collected from three MOGs that contain in total over 10 years of behavioral history for millions of players and matches. We compare social relationships in MOGs across different game genres and with regular online social networks like Facebook. Taking match recommendation as an example application of our model, we propose SAMRA, a Socially Aware Match Recommendation Algorithm that takes social relationships into account. We show that our model not only improves the precision of traditional link prediction approaches, but also potentially helps players enjoy games to a higher extent.

Additional Key Words and Phrases: Multiplayer Online Games, social relationship, user interaction, graph model

### 1. INTRODUCTION

Multiplayer Online Games (MOGs) are games in which multiple players can play in the same online game environment at the same time. They have attracted hundreds of millions of people world-wide who communicate, interact, and socialize with each other through gaming. They represent a large economic sector that covers an entire ecosystem of entertainment products, and that is worth billions of US dollars world-wide. Some MOGs, for example Defense of the Ancients (DotA) and StarCraft II, have featured in several tournaments with wide appeal to gamers and game-watchers, such as the World Cyber Games (WCG) and the Electronic Sports World Cup (ESWC). From the vast user base of MOGs, rich social relationships emerge that are useful for improving gaming services such as match recommendation and game population retention.

---

<sup>1</sup>Jia is currently working in the Information and Electrical Engineering Department at the China Agricultural University.

---

Author's addresses: A.L. Jia, S. Shen, A. Iosup, and D.H.J. Epema, Department of Software and Computer Technology, Delft University of Technology; R. van de Bovenkamp, and F. Kuipers, Department of Intelligent Systems, Delft University of Technology.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies show this notice on the first page or initial screen of a display along with the full citation. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, to republish, to post on servers, to redistribute to lists, or to use any component of this work in other works requires prior specific permission and/or a fee. Permissions may be requested from Publications Dept., ACM, Inc., 2 Penn Plaza, Suite 701, New York, NY 10121-0701 USA, fax +1 (212) 869-0481, or [permissions@acm.org](mailto:permissions@acm.org).

© YYYY ACM 1556-4681/YYYY/01-ARTA \$15.00

DOI : <http://dx.doi.org/10.1145/0000000.0000000>

Thus, it is important to understand the social relationships in MOGs, and this constitutes the purpose of this paper.

Different from single-player games that put players against program-controlled opponents, MOGs allow players to enjoy interactions with other human beings—they may compete with each other individually, they may work cooperatively as a team to achieve a common goal, they may supervise activities of other players, or they may engage in a game genre that incorporates any possible combination of these types of interactions. Often, these interactions provide players with a form of social communication, from which various types of social relationships may emerge that can be used for improving gaming services. For example, plain adversarial relationships in MOGs can be used to promote user activity—players who want to beat each other may motivate each other to stay longer as game customers.

An increasing number of social network analyses [Newman 2003] use graphs to represent user relationship, including *explicit* relationships like friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009], and *implicit* relationships like user interactions [Wilson et al. 2009], [Viswanath et al. 2009], [Leskovec et al. 2005], [Xiang et al. 2010], [Liu et al. 2012]. Often, graphs are extracted from one or multiple *snapshots* of the network based on a *single, domain-specific*, and usually *threshold-based* rule for mapping relationships to links. Taking Facebook as an example, the mapping rule can be that any two users who have exchanged messages for more than ten times are mapped to two nodes connected by one link in the graph. For three reasons this traditional approach is insufficient to obtain clear social relationships and their evolution from an MOG. First, network snapshots can be generated in various ways, for example, with or without considering the history before the last snapshot. However, the influence of different snapshot generation methods on the inferred relationship evolution has only been partially investigated before [Viswanath et al. 2009], [Leskovec et al. 2005], [Ribeiro et al. 2013], [Caceres and Berger-Wolf 2013], [Krings et al. 2012], and a detailed comparison between these methods is needed. Secondly, gaming involves relationships in various domains that normally do not exist in regular social networks, for example, winning together and competing with each other. To infer user relationships in MOGs, these domains need to be carefully examined and compared. Thirdly, the impact of various graph extraction rules and thresholds employed has received relatively little attention [Choudhury et al. 2010], and a thorough analysis on how they influence the structure of the resulting graphs is needed.

To tackle the above issues, in this paper, we propose a graph model that is able to capture social relationships of a variety of types and strengths. We apply our model to real world datasets to analyze social relationships and their evolution in MOGs. As it turns out, our model is able to identify important relationships in MOGs, including “wingmen” that are very likely to play together in the future and adversary relationships that are useful for population retention. By investigating network evolution from different perspectives, our model also demonstrates how an MOG can still exhibit growth while its player activity actually declines. Taking match recommendation as the example, we demonstrate how to apply the social relationships revealed by our model to improve gaming services. Our results show that our model not only improves the precision of traditional link prediction approaches, but also potentially helps players enjoy games to a higher extent. We summarize our main contributions as follows:

- (1) We collect, use, and offer public access to datasets representative for three popular MOG genres. The datasets contain in total 10 years of behavioral history for 1,120,049 players and 2,248,045 matches (Section 3). The datasets are publicly available through the Game Trace Archive (<http://gta.st.ewi.tudelft.nl/>).

- (2) We propose a graph model to analyze social relationships in MOGs, based on which we study the influence of relationship strength on the extracted graphs (Section 4), we analyze the mutual influence of friendship and interactions (Section 5), and we demonstrate user behavioral changes and the resulting network evolution in MOGs (Section 6).
- (3) Based on the social relationships revealed by our model, we propose SAMRA, a Socially Aware Match Recommendation Algorithm that we compare with traditional link prediction approaches (Section 7).

## 2. PROBLEM STATEMENT

In this section, we first introduce Multiplayer Online Games (MOGs), matchmaking, and social relationships in MOGs. Then, we state the research questions we study in this paper.

### 2.1. An overview of MOGs

Multiplayer Online Games are games in which multiple players can play in the same online game environment at the same time. Players in MOGs often control in-game avatars and, individually or team-wise, they try to conquer the opposite side's territory. Within MOGs a variety of game genres exist. Throughout this paper, we consider three popular game genres, namely Real Time Strategy (RTS) games exemplified by StarCraft II, Multiplayer Online Battle Arenas (MOBAs) exemplified by DotA, and Massively Multiplayer Online First-Person Shooter (MMOFPS) games exemplified by World of Tanks.

Different game genres often have different match scales that specify a limit on the number of players per team, and in-game targets that reflect the design emphasis of the game genres. Regarding the match scale, DotA requires exactly 5 players per team (we indicate this by "5v5-player"), whereas StarCraft II and World of Tanks allow players to form teams of different sizes, with a maximum of 8 and 15 players per team, respectively. Regarding in-game targets, StarCraft II asks players to balance strategic and tactical decisions, often every second, while competing for resources with other players. DotA provides the opportunity for teams of players to confront each other on a map and try to conquer the opposite side's main building. And World of Tanks, although fast-paced, tests the tactical team work of players disputing a territory. Intuitively, from StarCraft II to DotA and further to World of Tanks, the requirement of team cooperation increases. As a consequence, we observe an increasing trend in their match scale—92% of the matches in our StarCraft II dataset are 1v1-player, all matches in DotA are exactly 5v5-player, and 98% of the matches in our World of Tank dataset are 15v15-player.

Seeing MOGs in a broader perspective, they are online social networks in which users socialize with each other through gaming. Here, we classify online social network into two categories, viz. *socializing-driven* and *target-driven* networks. Different from typical socializing-driven networks like Facebook, where users mainly join to socialize with their online and offline friends, MOGs are target-driven and their users *primarily* join and interact for a particular target, i.e., games. Other typical target-driven networks include YouTube, Flickr, and Meetup, where users interact through co-commenting on the same video, co-viewing the same photo, and co-participating in the same event. As in many other online social networks, users in MOGs may develop various social relationships.

It should be noted that our classification here is loose: it often happens that users in target-driven networks also seek to socialize with others. Nevertheless, we still use this classification to distinguish the premier purposes of different online social networks.

As a matter of fact, we observe interesting patterns based on this classification (as we will show later in Section 5.2).

## 2.2. Matchmaking in MOGs

As we focus in this paper on understanding the social relationships emerging from MOGs, it is important to know how matchmaking works, i.e., how users are paired/grouped into the same game. Normally, different matchmaking methods are employed by MOG communities. In this paper, we consider four MOG communities, i.e., two DotA communities, Dota-League and DotAlicious, one StarCraft II community, and one World of Tanks community (details of these communities will be introduced later in Section 3.1). We introduce their matchmaking methods in turn as follows.

In Dota-League, players who want to play a match first join a waiting queue. When there are 10 or more players in the waiting queue, the matchmaking algorithm will form teams (each with five players) that are balanced in terms of the skill levels of the players. Although this matchmaking algorithm enforces balanced matches, it does not take into account the social relationships of the players. As a consequence, we observe that 41% of the games in Dota-League are aborted at the very beginning of the match. Because quitting at the start of the game could be the outcome of players expressing their disagreement with the matchmaking system's choice, we omit these 41% games in our later analysis and assume that the formations in the remaining games are according to players' satisfaction.

In DotAlicious, each game server has a number of open matches waiting for players to join, and each arriving player can select which match to join and on which team. For StarCraft II, users can choose either to organize games by themselves (the so-called *custom game*), or to be assigned by the community to games with other players with similar skill levels (the so-called *ladder game*). World of Tanks uses a similar matchmaking method as StarCraft II.

To sum up, under the above matchmaking methods, users can choose their teammates and opponents freely, either by organizing the games directly, or by quitting playing with unintended players. Therefore, the gaming experiences in the four MOG communities we consider are suitable proxies for inferring spontaneous social relationships.

## 2.3. Social relationships in MOGs

Social relationships in MOGs can be *explicit* or *implicit*. Explicit relationships are formed on players' own initiative, for example, when players personally establish friendships with others, or join a clan (a self-organized group of players who often form a league and play on the same side in a match). Implicit relationships, on the other hand, are formed passively by players, for example, through interactions.

Explicit relationships are precise, but they are not sufficient to capture various social relationships in MOGs. One reason is the prosocial emotions involved through gaming, for example, vicarious pride and happy social embarrassment [McGonigal 2011]. Another reason is the rareness of explicit relationships like friendship. When explicit relationships are not enough, implicit relationships, such as user interactions, provide the supplementary information for inferring social relationships. For example, we observe that in Dota-League, less than 10% players have explicitly identified more than 10 friends, while 50% players have played repeatedly and possibly regularly with more than 50 players.

In MOGs, users interact with each other in various ways, for example, by joining the same match, by playing on the same/opposite sides, and by winning/losing together. These interactions are straightforward and yet important: joining the same match is a precondition for interaction, playing on the same/opposite side indicates a posi-

tive/adversarial relationship, and winning/losing together may impact player attitude and their future team formation. These interactions will help us to infer social relationships in MOGs.

## 2.4. Research questions

In this paper, we answer the following five research questions:

(i) *How can we infer social relationships from MOGs?*

User relationships provide important information for improving services in online social networks. However, most previous work is based on explicit relationships like friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009] or simply treats various types of interactions equally [Wilson et al. 2009]. In contrast, we propose a graph model that takes a variety of types of interactions into account.

(ii) *How are MOGs different across game genres and from regular online social networks at the structural level?*

This question helps us to understand the differences between social networks. In this paper, we apply our model to real-world data collected from three MOGs. We then compare our results with previous studies on regular online networks [Wilson et al. 2009], [Viswanath et al. 2009], [Mislove et al. 2007], [Liu et al. 2012], including Facebook, YouTube, LinkedIn, and Meetup.

(iii) *How are networks representing explicit and implicit relationships in MOGs related?*

The correlation between explicit and implicit relationships is useful for relationship prediction. For example, we can predict potential interactions of a user based on his current friendships with others. We answer the above question by comparing properties of the graphs representing friendship and interactions.

(iv) *How do MOGs evolve over time?*

Network evolution reflects general user activity change, and provides important information for system operation. Previous evolution models are mostly based on friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009]. A few models [Viswanath et al. 2009], [Leskovec et al. 2005] include user interactions, but they only consider one perspective on network evolution. In contrast, we investigate network evolution from different perspectives by demonstrating how an MOG can still exhibit growth while its player activity actually declines.

(v) *What are possible applications of our model?*

We take match recommendation as the example to study the application of our model to gaming services. Good match recommendation algorithms improve user experience and hence the commercial value of MOGs, but they are often neglected or designed in a casual way. To tackle this issue, we propose a socially aware algorithm based on our model, and we show that our model improves the quality of match recommendation.

## 3. A MODEL FOR MOGS

In this section, we first introduce the datasets we collected from three MOGs. Then, we propose a graph model for analyzing social relationships in MOGs.

### 3.1. Datasets of MOGs

Players in MOGs are often loosely grouped into large communities. Each of these communities operates its own game servers and provides various gaming services, such as matching players to games, maintaining lists of match results and user profiles, and publishing player ranking information. We have collected datasets from four MOG communities, i.e., two DotA communities, Dota-League and DotAlicious, one StarCraft II community, and one World of Tank community.

Table I. Dataset statistics of four MOG communities. Game genre and team size are as introduced earlier in Section 2.

Community	Game genre	Team size	No. of players	No. of matches	Obtained history
Dota-League	MOBAs	exactly 5v5	61,198	1,470,786	2008.11 - 2011.7
DotAlicious	MOBAs	exactly 5v5	62,495	617,069	2010.4 - 2012.2
StarCraft II	RTS	mostly 1v1	83,199	85,532	2012.3 - 2013.8
World of Tanks	MMOFPS	mostly 15v15	913,157	74,658	2010.8 - 2013.7

In these communities, each user possesses a user profile page which shows his friend list and clan membership. Each match has a match page that shows the start and end times of the match, the player list, and the result of the match (winning team, draw, or abort). We have crawled each of these user profiles and match pages at least twice, to reduce the effect of possible temporary unavailability and traffic shaping of the website. To sanitize the data, we have filtered out matches with zero duration. In the end, we have obtained the *full history* of the four MOG communities and in total four types of datasets: (i) the friendship dataset from Dota-League, (ii) the clan membership dataset from DotAlicious, (iii) the user skill level datasets from Dota-League and DotAlicious, and (iv) the match datasets for all four communities. Statistics of these datasets can be found in Table I.

In general, the two DotA communities, Dota-League and DotAlicious, achieve populations of similar sizes. Because Dota-League has been operated longer than DotAlicious, it has more matches. World of Tanks achieves the largest population but the fewest matches. We believe this is because it has the largest match scale: most matches in World of Tanks are 15v15-player, whereas matches in DotA and StarCraft II are 5v5-player and (mostly) 1v1-player. A more detailed description of the datasets can be found in our previous work [van de Bovenkamp et al. 2013] and [Iosup et al. 2014].

### 3.2. Graph Models for MOGs

Following social network analysis, we use graph-based models to represent user relationships in MOGs. We propose the following two types of graphs, in both of which the nodes represent the players:

*3.2.1. Friendship graph.* In the *friendship graph* a link between two nodes represents the friendship between the corresponding players. Friendship graphs are undirected and unweighted.

*3.2.2. Interaction graph.* Following previous work on Facebook [Wilson et al. 2009], in an *interaction graph* a link between two nodes represents interactions, in terms of games, between the corresponding two players. Unlike in [Wilson et al. 2009] where all interactions are assumed to be homogeneous, we consider five types of interactions and we extract five interaction graphs as follows:

- (1) SM: two players present in the *Same Match*.
- (2) SS: two players present on the *Same Side* of a match.
- (3) OS: two players present on the *Opposite Sides* of a match.
- (4) MW: two players who *Won* together in a match.
- (5) ML: two players who *Lost* together in a match.

Interaction graphs are undirected and unweighted; we do not use link weight to capture the interaction strength. Instead, we map interactions to links by applying a threshold-based rule, and only interactions with enough strength to pass the thresholds will be included in the graph. We consider two *mapping thresholds*: the period  $t$  of effect for a user interaction, and the minimum number  $n$  of interactions that need to have occurred between two users for a relationship to exist. For example, in an SM

graph with  $t$  equal to one week, and  $n$  equal to 10, a link between two players exists only if there is at least one week in which they have played at least 10 games together.

It is obvious that both a small value of  $t$  and a large value of  $n$  impose strong relationship constraints. Meanwhile, for the same values of  $t$  and  $n$ , there are fewer relationship constraints in the SM graph than in the SS and OS graphs, which in turn have fewer relationship constraints than the ML and MW graphs. Thus, by tuning  $t$  and  $n$  for our five interaction graphs, our model can capture relationships with various strengths. We explore this in more detail in Section 4.

The list of the five interaction graphs we propose here is not exhaustive and it can support more complex variations. For example, it can incorporate more specific interactions, such as playing against each other at least 10 times, in the winter, while located in the same country. It can also support more mapping thresholds. For example, opposite to  $n$ , we can specify a maximum number of interactions between two users for a relationship to exist. In this way, we can focus on moderately interacting user pairs, which often consist the majority of an MOG's population.

### 3.3. Graph metrics

To study the social relationships in MOGs, we compare their friendship and interaction graphs based on a number of graph metrics that are related to the degrees and paths between players. Specifically, we consider the following graph metrics:

**Network size ( $N$ ):** The number of non-isolated nodes in a graph;  $N_{LCC}$  represents the size of the Largest Connected Components (LCC) and  $N_{LCC}/N$  represents the fraction of nodes in the LCC.

**Number of links ( $L$ ):** The number of links in a graph;  $L_{LCC}$  represents the number of links in the LCC.

**Degree ( $d$ ):** The degree of a node is the number of its neighbors.

**The distance ( $h$ ):** The distance between two nodes is equal to the length of a shortest path between them.

**Diameter ( $D$ ):** The diameter is the largest distance between any two nodes.

**The clustering coefficient ( $C$ ):** The clustering coefficient (CC) of a node is equal to the fraction of pairs of its neighbors that are linked.

**Assortativity ( $\rho$ ):** Assortativity is the average Pearson Ranking Correlation Coefficient (PRCC) of degree between pairs of connected nodes. In brief, PRCC measures the linear dependence between two variables. Therefore, assortativity measures to what extent nodes link to other nodes with similar degrees.

## 4. USER RELATIONSHIP AND NETWORK STRUCTURE

The graph model proposed in Section 3 identifies relationship types and strength in MOGs by differentiating gaming relationships (SM, SS, OS, MW, and ML) and by using mapping thresholds ( $n$  and  $t$ ). In this section, we analyze the influence of the gaming relationship and the values of the thresholds on the structural properties of the graphs generated by our model. In general, the patterns we observe for the four MOG communities we consider are rather similar, and therefore, here we only show the results for Dota-League.

### 4.1. Influence of interaction strength: threshold $n$

Here, we set the period of effect  $t$  to  $\infty$  and we vary the minimum number of interactions  $n$  from 4 to 500. We choose this range for  $n$  because, as we will show later, it captures the important changes of the graph structure: when  $n$  increases from 4 to 500, the graph starts to dissolve from a giant connected component to a number of relatively small connected components. For every value of  $n$ , we generate a set of five



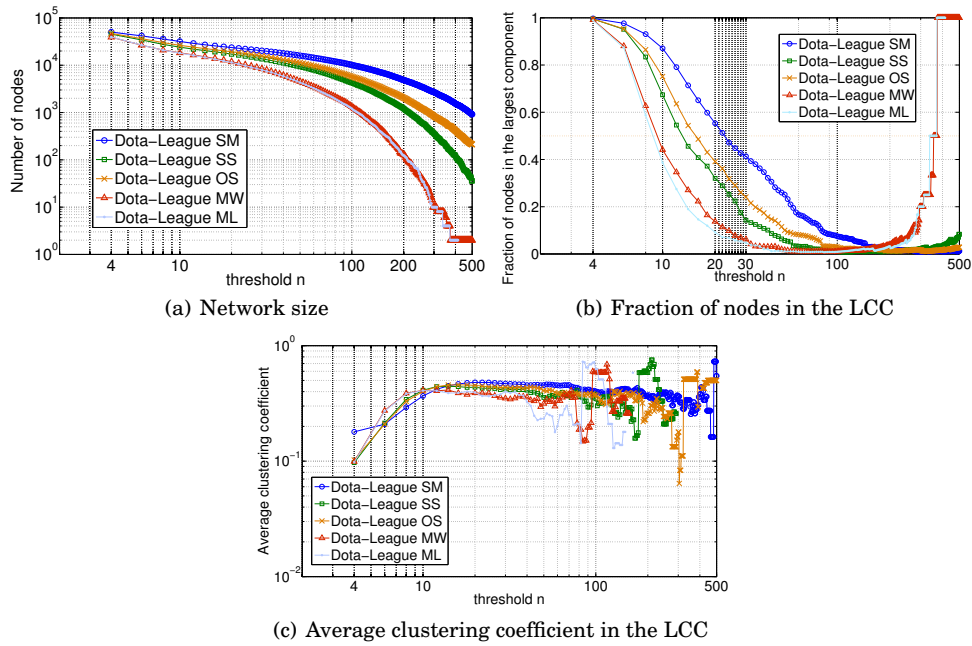


Fig. 1. The influence of the minimum number of interactions that need to have occurred between two users for a relationship to exist (threshold  $n$ ) on the network structure (the vertical axis has a logarithmic scale).

interaction graphs from the dataset of Dota-League, one for each gaming relationship. We show the properties of these interaction graphs in Fig. 1.

**Network size:** As shown in Figs. 1(a) and 1(b), for any of the interaction graphs, its network size and the *fraction* of nodes in the LCC drop quickly (near exponentially) as the threshold  $n$  increases, and after  $n$  increases to a very large value<sup>2</sup>, the fraction of nodes in the LCC starts to increase with  $n$ . Apparently due to the dramatically decreased network size, it becomes easier for the remaining nodes to be connected.

Intuitively, for pairs of players who play intensively with each other, the increase of  $n$  should not influence their links very much, since they have played far more than  $n$  games. We conjecture that less-intensively playing players form the less strongly connected fringe in the LCC, and that as  $n$  increases, their links are removed first and they are removed from the LCC—in other words, the core of the graph will be more strongly connected as  $n$  increases. This conjecture is confirmed by the following observation on the network connectivity.

**Network connectivity:** As shown in Fig. 1(c), for any of the interaction graphs, the average clustering coefficient increases with  $n$  until  $n \sim 20$ , stays stable after that, and starts fluctuating from  $n \sim 50$ . This result confirms our conjecture that as  $n$  increases, the less strongly connected fringe is removed and the LCC is getting more strongly connected. On the other hand, when  $n$  becomes very large, links from the core of the graph are also removed. The combination of these two forces makes the clustering coefficient unstable for large values of  $n$ .

**Default value of  $n$  for later analysis:** To avoid repeatedly exploring the mapping thresholds, here we choose the default value of  $n$  for our later analysis. On the one hand,  $n$  cannot be too large, otherwise it induces a very strong relationship constraint

<sup>2</sup>This turning point is different for different interaction graphs.

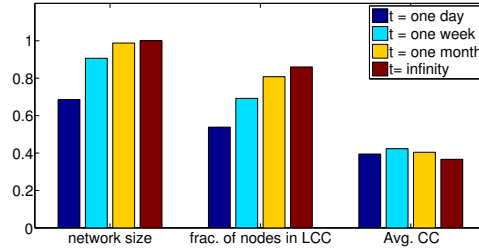


Fig. 2. The influence of the period of effect, i.e., the threshold  $t$  (network size normalized to the case of  $t = \infty$ )

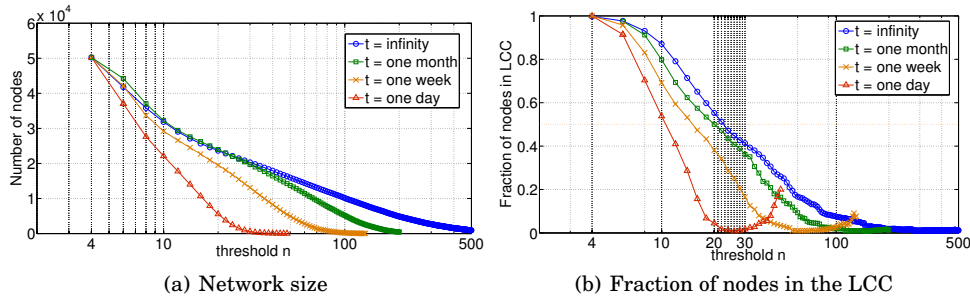


Fig. 3. The influence of the minimum number of interactions that need to have occurred between two users for a relationship to exist (threshold  $n$ ) and the period of effect (thresholds  $t$ ) on the network structure (the vertical axis in (b) and (c) has a logarithmic scale).

and the network will be overly fragmented. On the other hand,  $n$  cannot be too small either, otherwise it achieves a small clustering coefficient and it will be difficult to differentiate strong user relationships from occasional or even random ones. Given these considerations, we choose  $n = 10$  for Dota-League, where for any of the interaction graphs at least half of the nodes are in the LCC and the average clustering coefficient is almost at its maximum, which guarantees a strongly connected LCC with a reasonable size. Following the same rules, we choose  $n = 10$ , 2, and 4 as the default values for DotAlicious, StarCraft II, and World of Tanks, respectively.

#### 4.2. Influence of interaction type

As shown in Fig. 1, for the same value of  $n$ , the network size and the fraction of nodes in the LCC in the SM graph are larger than those in the OS and SS graphs, which in turn are larger than those in the MW and ML graphs. When  $n$  increases, these metrics drop faster in the MW and ML graphs than in the OS and SS graphs, which in turn drop faster than in the SM graph. One simple explanation is that there are fewer links in graphs extracted using stronger relationship constraints (e.g., MW and ML, or OS and SS compared to SM), and thus removing links from them breaks down the graph more quickly than graphs that are extracted using less restrictive relationship constraints.

#### 4.3. Influence of the period of effect: threshold $t$

Given the similarity in the network structure of the five interaction graphs we model, here we focus only on the SM graph. We set  $n = 10$  and vary  $t$  from one day, to one week, one month, and infinity. The results are shown in Fig. 2. With an increasing value of  $t$ , the relationship constraints get less strict, and so the network size and the fraction

of nodes in the LCC are increased, although compared to Fig. 1, they are not increased as quickly as when we decrease  $n$ . Meanwhile, the average clustering coefficient does not change much with  $t$ . These results indicate that  $n$  has a higher influence on the network structure than  $t$ , and we compare the influence of  $n$  and  $t$  in the following section.

#### 4.4. Comparison of the influence of $n$ and $t$

As shown in Figs. 3(a) and 3(b), for a fixed value of  $n$ , decreasing  $t$  can at most reduce the network size by 20,000 and the fraction of nodes in the LCC by 50% (when  $n = 20$ ), whereas for a fixed value of  $t$ , decreasing  $n$  can easily reduce the network size by over 30,000 and the fraction of nodes in the LCC by over 80%.

Together with the results from the previous sections, we conclude that when the constraints for a relationship to exist get stricter, the graphs representing those relationships become more disconnected, but the connection between nodes in their LCCs becomes stronger. Further, increasing  $n$  and decreasing  $t$ —two methods to extract more strict relationships—have similar influence on network structure, with increasing  $n$  having a stronger effect.

### 5. FRIENDSHIP AND INTERACTION GRAPHS

In this section, we analyze the similarities, the differences, and the correlation between the friendship and interaction graphs of the four MOG communities we consider. To generate the interaction graphs, we use the default values for the mapping thresholds we chose in Section 4, i.e., the period of effects for interactions  $t$  is set to  $\infty$ , and the minimum number of interactions  $n$  is set to 10, 2, and 4 for Dota-League and DotAlicious, StarCraft II, and World of Tanks, respectively. To compare different MOGs at the same activity level, we also test  $n = 10$  for StarCraft II and World of Tanks. An overview of the values of the graph properties is presented in Table II. In general, we find that both the friendship and interaction graphs of the four MOG communities we consider exhibit the small-world property, that friendship has a positive influence on user interactions, and that MOGs of game genres have different user interaction patterns.

#### 5.1. The small-world property

As shown in Table II, both the friendship and interaction graphs have relatively small average hop counts and rather high average clustering coefficients, indicating that they possess small-world properties [Milgram 1967], [Watts and Strogatz 1998] rather than the properties expected of random graphs<sup>3</sup>.

#### 5.2. Degree distribution

Instead of finding accurate degree distributions, we are more interested in understanding the difference in the degree distributions of the graphs representing different relationships. We use a power-law distribution as the comparison baseline, since node degrees in many social networks are found to follow this distribution [Mislove et al. 2007], [Wilson et al. 2009], [Liu et al. 2012]. We use the Maximum Likelihood Estimation method [Clauset et al. 2009] to perform power-law curve fitting for node degrees in the friendship and interaction graphs, and we compare whether one is heavier, similar, or lighter-tailed than another.

Fig. 4 shows the results of Dota-League. The straight dashed lines in these figures are the fitted power-law distributions. We see that the degree distribution of Dota-

<sup>3</sup>Random graphs have clustering coefficient equal to  $K/N$ , where  $K$  represents the average node degree and  $N$  represents the number of nodes.

Table II. Properties of friendship and interaction graphs extracted from four MOG communities. The metrics we present are the number of nodes  $N$ , the number of nodes in the largest connected component  $N_{LCC}$ , the number of links  $L$ , the number of links in the largest connected component  $L_{LCC}$ , the average hop count  $\bar{h}$ , the diameter  $D$ , the average clustering coefficient  $\bar{C}$ , and the assortativity  $\rho$ .

	DotA-League ( $n = 10$ )						DotAlicious ( $n = 10$ )				
	SM	OS	SS	ML	MW	FR	SM	OS	SS	ML	MW
$N$	31,834	26,373	24,119	18,047	18,301	53,062	31,702	11,198	29,377	22,813	21,783
$N_{LCC}$	27,720	19,814	16,256	6,976	8,078	34,217	26,810	10,262	20,971	10,795	13,382
$L$	202,576	85,581	62,292	30,680	33,289	368,854	327,464	92,010	108,176	43,240	54,009
$L_{LCC}$	199,316	79,523	54,186	17,686	21,569	301,586	323,064	91,354	99,063	29,072	44,129
$\bar{h}$	4.42	5.40	6.30	8.09	7.67	8.1	4.24	3.97	5.3	6.80	5.95
$D$	14	21	24	28	26	26	17	12	19	20	22
$\bar{C}$	0.37	0.40	0.41	0.41	0.41	0.3	0.43	0.27	0.47	0.47	0.49
$\rho$	0.13	0.26	0.25	0.27	0.28	0.22	0.08	0.01	0.25	0.27	0.29
	StarCraft ( $n = 10$ )					World of Tanks ( $n = 10$ )					
	SM	OS	SS	ML	MW	SM	OS	SS	ML	MW	
$N$	907	611	314	95	212	4,340	477	4,251	561	1,824	
$N_{LCC}$	31	22	24	9	14	129	118	122	66	57	
$L$	748	404	327	85	200	9,895	3,253	6,543	1,564	2,923	
$L_{LCC}$	58	21	44	13	24	2,329	1,243	1,160	519	473	
$\bar{h}$	1.88	4.04	1.90	1.64	1.74	1.88	2.00	2.18	2.080	1.78	
$D$	2	8	3	2	2	6	3	5	4	3	
$\bar{C}$	0.58	0	0.70	0.65	0.65	0.79	0.10	0.78	0.88	0.87	
$\rho$	-0.46	-0.45	-0.42	-0.58	-0.53	-0.10	-0.12	-0.06	-0.03	-0.10	
	StarCraft ( $n = 2$ )					World of Tanks ( $n = 4$ )					
	SM	OS	SS	ML	MW	SM	OS	SS	ML	MW	
$N$	83,199	83,199	25,556	13,995	10,736	78,226	15,618	68,659	27,605	28,274	
$N_{LCC}$	68,335	68,335	6,221	565	2176	20,555	4,932	8,444	163	2,911	
$L$	156,941	125,371	32,489	15,671	13,513	212,103	81,225	124,600	52,697	58,532	
$L_{LCC}$	143,892	113,994	11,907	1,139	3,983	145,141	35,615	38,671	799	14,877	
$\bar{h}$	7.21	7.58	7.23	3.94	6.62	9.85	13.55	12.42	3.30	17.98	
$D$	29	29	19	12	18	31	35	39	9	47	
$\bar{C}$	0.31	0.05	0.79	0.89	0.82	0.56	0.01	0.61	0.71	0.71	
$\rho$	-0.07	-0.09	-0.12	-0.16	-0.17	0.06	0.09	0.17	-0.003	-0.14	

League’s friendship graph is power-law distributed, and those of its interaction graphs are lighter-tailed than a power-law distribution. We have found similar results for interaction graphs of DotAlicious, StarCraft II, and World of Tanks.

**Comparison with regular online social networks:** In Section 2, we have classified online social networks, including MOGs, into *socializing-driven* and *target-driven* networks. Here, we compare the differences in their degree distributions.

For friendship graphs, the degree distributions in socializing-driven networks such as Facebook and Orkut do not follow power-law distributions [Mislove et al. 2007], [Wilson et al. 2009]. In contrast, the degree distributions in many target-driven networks are found to follow power-law distributions, as shown above for Dota-League, and in [Mislove et al. 2007] for Flickr, LiveJournal, and YouTube.

For interaction graphs, in socializing-driven networks such as Facebook, node degrees are significantly fitted by power-law distributions [Wilson et al. 2009], whereas for target-driven networks they are not: depending on the number of users involved in a target, they can be lighter-tailed or heavier-tailed than power-law distributions. For example, DotA limits the number of players per game to 10, whereas Meetup allows thousands of people participate in the same event. As a consequence, we observe that degree distributions of Dota-League’s interaction graphs are lighter-tailed than power-law distributions, and previous work [Liu et al. 2012] shows that for Meetup it is heavier-tailed than a power-law distribution.

Moreover, in Dota-League, we observe smaller probabilities for high degree nodes going from the SM graph to the SS and OS graphs, and further to the MW and ML

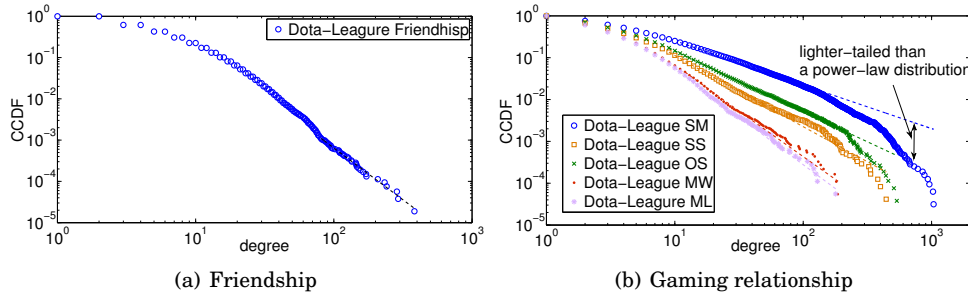


Fig. 4. The degree distributions of the friendship and interaction graphs of Dota-League. Straight dashed lines show the fitted power-law distributions.

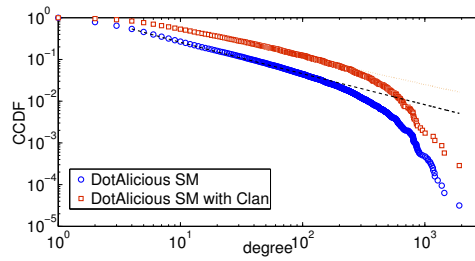


Fig. 5. The degree distributions of the interaction graphs of DotAlicious. Straight dashed lines show the fitted power-law distributions.

graphs. We believe that as the relationship constraints get more restricted (i.e., from SM to SS and OS, and further to MW and ML), fewer player pairs will pass threshold  $n$  and establish links between them. A similar phenomenon has been found in the interaction graphs extracted from Facebook [Wilson et al. 2009] and LiveJournal [Mislove et al. 2007].

### 5.3. The correlation of friendship and interactions

We use the Pearson Ranking Correlation Coefficient (PRCC) [Rodgers and Nicewander 1988] to measure the correlation between the number of friends and the number of interactions of a player in MOGs. In brief, PRCC measures the linear dependence between two variables. We find that in Dota-League, there is a positive correlation between the number of friends in the FR (Friendship) graph and the number of interactions in the SM (Same Match), SS (Same Side), OS (Opposite Sides), MW (Matches Won together), and ML (Matches Lost together) graphs, achieving a PRCC of 0.3838, 0.4356, 0.4271, 0.4850, 0.5192, respectively. And in DotAlicious, as shown in Fig. 5, the node degree in general is higher when the SM graph consists of players with clan membership, compared to the case when all players are considered. Note that a clan is a self-organized group of players who often form a league and play on the same side in a match. These results indicate that players with strong explicit social relationships, like friendship and clan membership, tend to play more games.

Further, we find from Table II that for Dota-League (DotAlicious), its OS has similar (smaller) network size than its SS graph, i.e., compared to players in Dota-League, players in DotAlicious are more prone to play on the same side. Possibly due to the clan feature in DotAlicious, players who often play on the same side are committed to each other.

#### 5.4. The influence of game genre

So far, taking DotA, StarCraft II, and World of Tanks as the examples, we have shown a structural similarity in MOGs: naturally emerging social structures centered around highly active players. Nevertheless, as introduced in Section 2, these MOGs represent different game genres, and therefore, differences in their user relationships are expected. In this section, we further investigate these game genres.

**Alone or together?** As shown in Table II, for the same threshold  $n = 10$ , while tens of thousands of players have played in StarCraft II and World of Tanks, their interaction graphs only contains a few thousands of players, and the fractions of nodes in the LCC are extremely small. Apparently, StarCraft II and World of Tanks have much fewer players that would engage in repeated games with same players. Further, as shown in Table I, StarCraft II and World of Tanks have larger numbers of players but smaller numbers of matches compared to Dota-League and DotAlicious, which implies lower user activities in these two communities.

The above result also indicates that  $n = 10$  is so strict that it filters out most player pairs in StarCraft II and World of Tanks. In the following sections, we use the default values as we chose earlier in Section 4, i.e.,  $n = 2$  and 4 for StarCraft II and World of Tanks, respectively.

**Bonding or fighting?** Comparing the size of the SS and OS graphs, we see that players tend to play on the opposite side in StarCraft II (83,199 players in its OS graph versus 25,566 players in its SS graph), on the same side in World of Tanks (15,618 vs. 68,659), and with no strong preference on the playing side in Dota-League (26,373 vs. 24,119). Players in DotAlicious tend to play on the same side (29,377 vs. 11,198), but the tendency is not as strong as in World of Tanks. We believe this is due to the clan feature provided by DotAlicious, rather than the game genre. Intuitively, from RTS to MOBAs and further to MMOFPS games, the requirement of team cooperation increases, and therefore players are more likely to maintain an SS relationship with each other.

**Balance or challenge?** Assortativity measures to what extent players link to other players with similar node degree. As node degree represents player popularity, a positive assortativity indicates that players with similar popularity often play together, and a negative assortativity indicates the opposite. We find that DotA and World of Tanks always achieve positive assortativities for their SM, SS, and OS graphs, whereas StarCraft II always achieves negative ones.

The above result suggests that in games where individualistic skill prevails, for example RTS games exemplified by StarCraft II, players tend to seek challenges by playing with popular players. Intuitively, this is an effective way to improve player skills, since popular players have played with many others so that they potentially attained high skill levels. In fact, we do find a positive correlation (with a Pearson Correlation Coefficient of 0.6191) between node degree and player skill level in Dota-League, where the skill level is defined based on the fraction of matches a player has won. In StarCraft II, teammates and opponents are either chosen by players themselves, or assigned by the community based on the similarity of skill levels (see also Section 2.2). Therefore, we attribute the proneness of playing with highly skilled players in StarCraft II to a player's willingness to seek challenges.

Overall, our analysis in this section shows that different game designs have different influences on the social relationships emerging among players. For one application, game designers and MOG community administrators could use our analysis as a reference to adjust their designs and to maneuver, or manipulate, their players. For example, our analysis shows that players in StarCraft II tend to seek challenge and compete with each other (by playing on the opposite side). Administrators of MOG communities

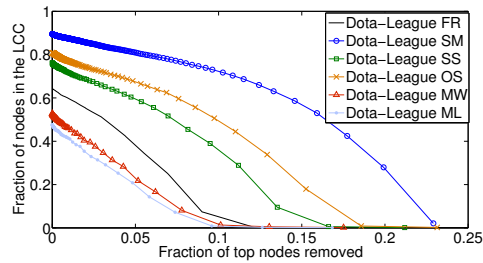


Fig. 6. The fraction of nodes remaining in the LCC as a function of the fraction of the top nodes that are removed.

that are similar to StarCraft II can therefore create a competitive environment, e.g., by organizing some tournaments or publishing player ranks, to promote the activity level of their players and potentially achieve a higher commercial revenue.

We also conjecture that these game genres vary in the extent to which players socialize. However, our data do not allow an analysis on this topic. To do so, we will need extra information, like how many private messages have exchanged between players. We consider this an interesting topic for future work.

### 5.5. The importance of top players

In this section, we analyze how the network is connected when top players are removed gradually. We compare two types of top players, i.e., the ones who have many friends and the ones who have played games with many others, which are identified by their node degrees in friendship and interaction graphs. As Dota-League is the only community for which we have obtained both friendship and interaction information, we use it as the example.

Previous work [Mislove et al. 2007] has shown that, in some online social networks where the friendship graphs have power-law distributed node degrees (for example YouTube), removing top nodes quickly breaks the whole graph apart. Consistent with this observation, as shown in Fig. 6, the FR graph of Dota-League also breaks down quickly as more top nodes are removed: with 10% top nodes removed, the fraction of nodes in the LCC is almost decreased to zero. Further, we observe that interaction graphs also break down quickly when top nodes are removed. Similar observations in other online social networks, but not in online games, have been found in [Jiang et al. 2013].

The above results indicate that top players are important for keeping the connectivity and holding the whole community together. Community administrators can adopt special policies to keep the activity of these top players, and therefore, the activity of the community.

### 5.6. Triadic closure

A closed triad is a group of three nodes who are connected with each other. In psychology, it has been shown that triadic closure is more likely to happen with positive rather than negative relationships, i.e., a friend of my friend is likely to be a friend whereas an enemy of my enemy is less likely to be an enemy [Heider 1946], [Cartwright and Harary 1956]. In this section, we test whether this phenomenon also happens in MOGs.

The pro-social and the enmity relationships are strongly expressed in gaming, whereas the latter may be repressed in some real-world settings, especially professional. Therefore, here we consider playing on the same side (SS) as a positive relationship (represented by “+” in Fig. 7) and playing on the opposite side (OS) as a

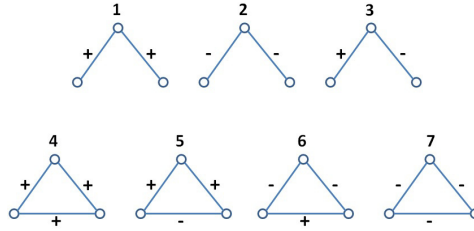


Fig. 7. The 7 classes of triads and their ids. Triads 1 to 3 are connected but not closed, and triads 4 to 7 are closed. The transition of triad 1 to triad 4 represents an example of triadic closure.

Table III. The fraction of triads that are closed in the SS and OS graphs, respectively, of four MOG communities.

	Dota-League	DotAlicious	StarCraft II	World of Tanks
SS graph	9.24%	6.62%	2.83%	64.68%
OS graph	7.52%	6.89%	0.17%	3.03%

Table IV. The number and the percentage of class 4 to class 7 triads (as shown in Fig. 7) in the original signed graph ( $N_{\Delta}$ ) and in the random signed graph ( $N_{\Delta}^{rand}$ ), respectively. The results shown in the bold font represent cases that follow the social balance theory.

	Triad id	Class 4 triad	Class 5 triad	Class 6 triad	Class 7 triad
Dota-League	$N_{\Delta}$	297,495 (8.98%)	<b>1,078,516 (32.56%)</b>	<b>1,341,609 (40.50%)</b>	594,991 (17.96%)
	$N_{\Delta}^{rand}$	353,745 (10.68%)	<b>1,172,524 (35.40%)</b>	<b>1,302,035 (39.31%)</b>	484,307 (14.62%)
DotAlicious	$N_{\Delta}$	364,475 (9.37%)	<b>1,198,269 (30.82%)</b>	<b>1,592,434 (40.95%)</b>	733,104 (18.85%)
	$N_{\Delta}^{rand}$	548,411 (14.10%)	<b>1,514,709 (38.96%)</b>	<b>1,395,471 (35.89%)</b>	429,691 (11.05%)
StarCraft II	$N_{\Delta}$	<b>16,551 (10.12%)</b>	<b>5,344 (3.27%)</b>	<b>130,494 (79.72%)</b>	<b>11,216 (6.86%)</b>
	$N_{\Delta}^{rand}$	<b>2,060 (1.26%)</b>	<b>19,625 (12.00%)</b>	<b>66,291 (40.52%)</b>	<b>75,629 (46.23%)</b>
World of Tanks	$N_{\Delta}$	<b>1,217,812 (18.85%)</b>	<b>407,918 (6.32%)</b>	<b>4,677,089 (72.41%)</b>	<b>156,396 (2.42%)</b>
	$N_{\Delta}^{rand}$	<b>1,217,812 (12.5%)</b>	<b>3,653,436 (37.50%)</b>	<b>3,653,436 (37.50%)</b>	<b>1,217,812 (12.50%)</b>

negative relationship (represented by “-” in Fig. 7). In Fig. 7 we show the classes of triads that can happen in MOGs. The transition of triad 1 to triad 4 represents the triadic closure that may happen in the SS graph. Similarly, the transition of triad 2 to triad 7 represents the triadic closure that may happen in the OS graph.

In Table III we show the percentage of triadic closures in the SS and OS graphs of the four MOG communities we study. We see that Dota-League, StarCraft II, and World of Tanks achieve higher triadic closures in their SS graphs than in their OS graphs. This result confirms that triadic closure is more likely to happen among positive relationships, i.e., playing on the same side. For DotAlicious, we find similar triadic closures for its SS and OS graphs. One possible reason is that the clan feature provided in DotAlicious diminishes the significance of playing on the opposite side being a negative relationship. It remains for future work on other datasets to establish whether our conjecture is valid.

### 5.7. Social balance

The *social balance theory* reveals a phenomenon that is often observed in signed graphs, i.e., graphs with a “+” or “-” sign for each link. In this theory, a triad is defined as positive (balanced) if the product of the signs of its links is positive, and negative (unbalanced) otherwise. The *social balance theory* claims that balanced (unbalanced) triads in social networks should be over (under) represented compared to random graphs [Heider 1946], [Cartwright and Harary 1956], [Heider 1944]. In this section, we test the social balance theory based on our data.



For each of the four MOG communities, we first combine its SS and OS graphs into one signed graph, which we call the *original signed graph*. Then, we randomize the sign of each link in this signed graph to generate a *random signed graph*. Note that the random signed graph keeps the same fractions of “+” and “-” signs, and the same graph structure as the original signed graph. Therefore, our following analysis on social balance will not be influenced by the graph structure.

To test the social balance theory, we consider the four classes of closed triads (triads 4 to 7) as shown in Fig. 7. In Table IV, we show the number and the percentage of class 4 to class 7 triads in the original signed graph and in the random signed graph, represented by  $N_{\Delta}$  and  $N_{\Delta}^{rand}$ , respectively. The results in bold font represent cases that follow the social balance theory, i.e., balanced (unbalanced) triads in social networks should be over (under) represented compared to random graphs

We see that the social balance theory holds in most cases, except for triads 4 and 7 in Dota-League and DotAlicious. Recall that Table III shows that, compared to StarCraft II and World of Tanks, Dota-League and DotAlicious also have more similar percentages of triadic closure for their SS and OS graphs. These results suggest that while the social balance theory holds for most MOG communities, the significance of SS being a positive relationship and OS being a negative relationship varies across different MOG communities.

Szell *et al.* [Szell and S.Thurner 2012] have also observed a similar social balance phenomenon in an MOG named *Pardus*. Nevertheless, in their analysis, the positive and the negative relationships are identified by users, while in our analysis they are revealed explicitly by user interactions.

## 6. BEHAVIORAL CHANGE AND NETWORK EVOLUTION

In Sections 4 and 5 we have always considered the whole datasets. For Dota-League, it contains user interactions from November 2008 to February 2012. Obviously, players may change their interaction patterns and therefore, the network evolves over time. In this section, we study user behavioral change and network evolution in MOGs.

### 6.1. Two models for network evolution

In this section, we propose two models for analyzing network evolution in MOGs. Existing network evolution models are insufficient for our analysis, for two reasons. First, they are mostly based on friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009], whereas we consider more dynamic relationships, i.e., user interactions. Secondly, a few models [Viswanath et al. 2009], [Leskovec et al. 2005], [Merritt and Clauset 2013], [Ribeiro et al. 2013], [Caceres and Berger-Wolf 2013] do include user interactions, but they only consider one perspective on network evolution, whereas we differentiate and compare two types of network evolution, *network dynamics* and *network growth*.

Without this comparison, contradicting conclusions can be drawn from the incomplete analysis. For example, researchers in [Leskovec et al. 2005] have proposed a *densification law* which states that many social networks densify over time, with the number of edges growing super-linearly in the number of nodes. And researchers in [Merritt and Clauset 2013] contradicted this finding by showing that the friendship network they examined is non-densifying. The problem is that they have considered different types of network evolution. And as we will show later, depending on the evolution models, both densification and non-densification can happen even for the same network.

We define the *periodic graphs* of a network for a certain time duration as the sequence of graphs obtained by only considering those user interactions that have occurred in the successive periods of that duration—one can think of a periodic graph of

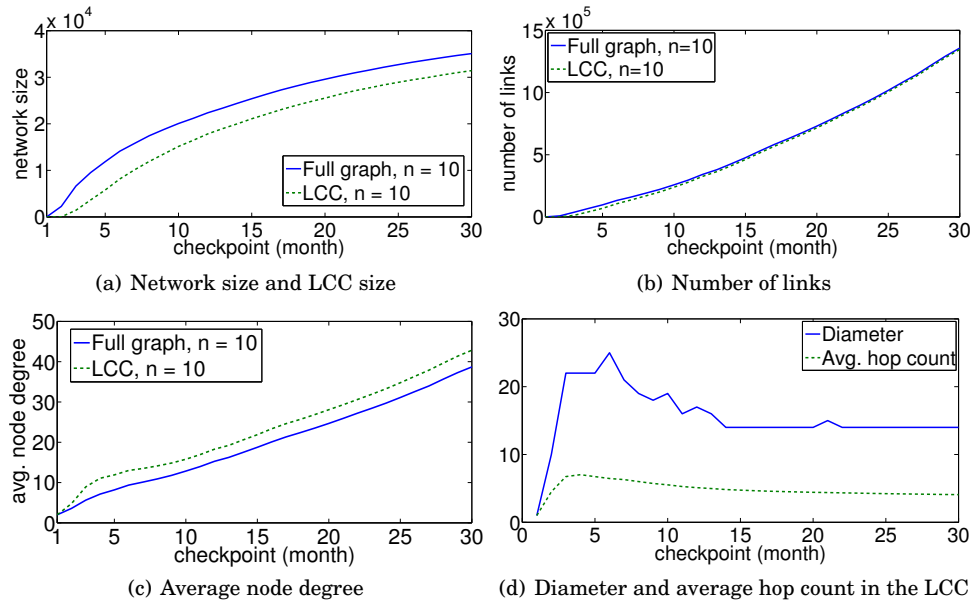


Fig. 8. Network evolution: cumulative graphs (checkpoint interval equal to one month,  $n = 10$ ).

a network as starting at the beginning of the corresponding period without any edges, and with only edges added for interactions that occur within the specific period. In contrast, the *cumulative graph* of a network up to a certain point in time has edges for all interactions that have ever occurred up to that time. We consider the periodic and the cumulative graphs to capture network dynamics and network growth, respectively. Under both models, we examine the network periodically based on a pre-defined *checkpoint interval* (the length of the period). At each checkpoint, we generate a periodic graph from the interactions that happened *within* the corresponding interval, and we generate a cumulative graph based on *all* the interactions that happened before that checkpoint.

Previous work on network evolution that comes closest to our analysis is [Krings et al. 2012], where the authors examined the effects of time window size (corresponding to the *checkpoint interval* in our approach) and placement (which decides whether it is a cumulative or a periodic graph in our approach) on the structure of aggregated networks. Nevertheless, they only consider one type of interaction, i.e., phone calls, and they assume a link exists between two users as long as they have interacted before, regardless of their interaction strength. In contrast, as introduced in Section 3.2, we consider different types of interactions, and we use threshold  $n$  (which reflects the interaction strength) to decide whether links should be added between players. We also analyze the influence of  $n$  on the network evolution models.

In our analysis, for each of the four MOG communities, we consider the above two models with three checkpoint intervals, i.e., one week, one month, and half a year, and two interaction thresholds, i.e.,  $n = 1$  and  $n = 10$ . In total, we generate 12 sets of graphs for each community (two types of evolution, three checkpoint intervals, and two thresholds), and each set consists of 5 interaction graphs (SM, SS, OS, MW, and ML graphs). In general, we observe similar patterns for the network evolution of the interaction graphs in these communities, and therefore, we only show the results for the SM graph of Dota-League in Figs. 8, 9, and 10.

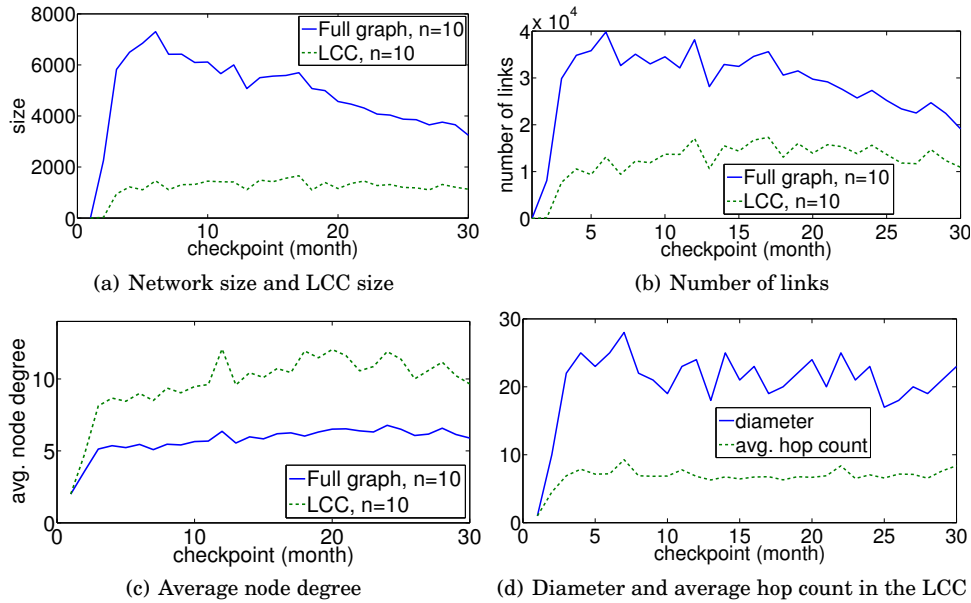


Fig. 9. Network evolution: periodic graphs (checkpoint interval equal to one month,  $n = 10$ ).

## 6.2. Network dynamics versus network growth

For the cumulative graphs, regardless of the checkpoint interval, the network size (Figs. 8(a)), the number of links (Fig. 8(b)), and the average node degree (Fig. 8(c)) increase over time, indicating that the whole network is getting denser. We also observe from Fig. 8(d) that after a short period of increase, the diameter of LCC actually decreases over time. Similar phenomena in network growth, i.e., *network densification* and *shrinking diameter*, have been observed in many other networks as well [Leskovec et al. 2005].

While the cumulative graphs seem to demonstrate the prosperity of the network, the periodic graphs show different, or even opposite trends in the network. We see from Figs. 9(a) and 9(b) that, after a short period of increase (within the first 5 months), the network size and the number of links in fact decrease over time, indicating that as time evolves, the whole network becomes less active. This is partially due to the decreasing popularity of Dota-League, which eventually led to its shut down in 2012. Meanwhile, we observe that the increase of the node degree and the decrease of the diameter over time are more obvious in the cumulative graphs (Fig. 8) than in the periodic graphs (Fig. 9).

The above results indicate that the understanding of network evolutions depends significantly on the network evolution model, and a clear definition of network evolution is crucial for understanding the network.

## 6.3. Committed early members

As shown in Fig. 10, for both evolution models, the first half year (i.e., 6 months and around 24 weeks) is a very special case. Compared to the rest of the data, it has the smallest network size and the smallest LCC size (Figs. 10(a) and 10(b)), yet it achieves the highest clustering coefficient (Figs. 10(c) and 10(d)). Intuitively, one would expect a higher clustering coefficient for the graph with a smaller network size, simply because there will be fewer nodes to choose at endpoints. However, we observe that in the

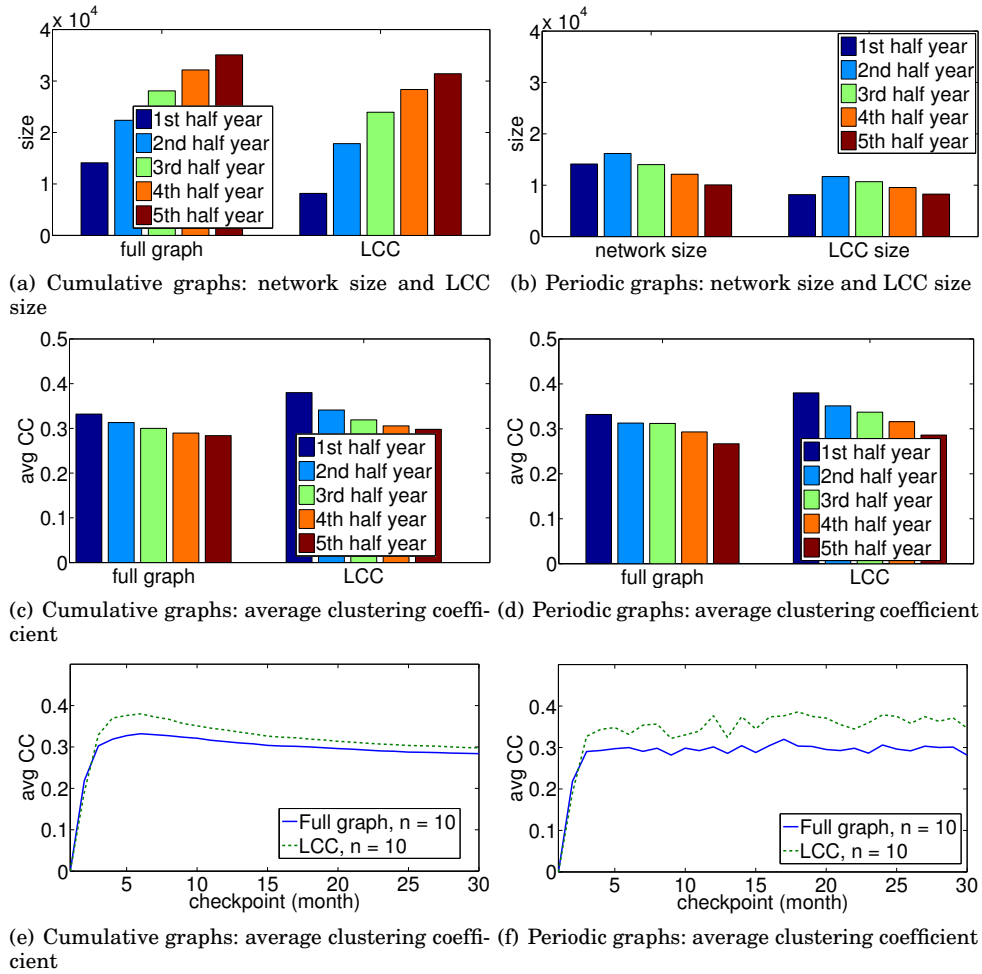


Fig. 10. Network evolution: the special case of the first half year (checkpoint interval equal to half year or one month,  $n = 10$ ).

first half year, both the network size (Fig. 9(a)) and the average clustering coefficient (Fig. 10(f)) increases with time. These results show that in the early days, though Dota-League had not attracted as many players as it later did, players were connected more closely then, than later. We conjecture that this phenomenon—in the early days, members are often more committed to the community—also happens in many other online and offline communities.

#### 6.4. The influence of threshold $n$

We have also tested the case of  $n = 1$  for our models using Dota-League SM graph as the example, and we observe a similar trend of network evolution as in the case of  $n = 10$ . As an example, Fig. 11 shows the network size and the average node degree for our periodic graph model. The case for  $n = 10$ , which requires a different scale, is depicted in Fig. 9. Similar to the case of  $n = 10$ , we observe that after the first 6 weeks, the network size decreases and the node degree increases slightly over time.

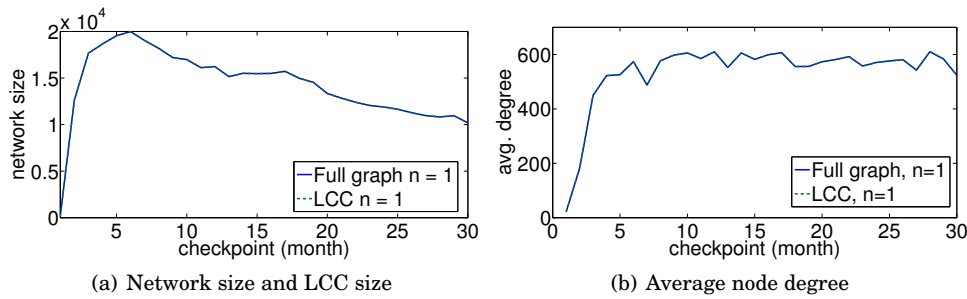


Fig. 11. Network evolution: the influence of threshold  $n$  (periodic graphs, checkpoint interval equal to one month,  $n = 1$ )

This result indicates that the threshold  $n$  influences mostly the scale of the evolution, but not the trend.

## 7. SOCIAL RELATIONSHIP AND MATCH RECOMMENDATION

Taking match recommendation as the example, in this section we study the implications of our model on gaming services.

### 7.1. Overview

Match recommendation in an MOG community predicts player pairs that are likely to form gaming relationships in the future, such as playing together, playing on the same and/or the opposite sides. Match recommendation often includes two types of predictions, i.e., *predicting new relationships* between players who previously had no relationships at all, and *predicting repeated relationships* between players who have formed the same relationships in the past.

Good match recommendation algorithms help improve user experience, and hence the commercial value of MOGs. However, they are often neglected or seem to have been only casually designed. For example, in Dota-League, players can only join a waiting queue, and, only when there are enough players, teams are formed considering the skill levels of the players in the game. Although this algorithm enforces balanced matches, it does not take into account the social relationships of players. As a possible consequence, we observe that 41% of the games in Dota-League are aborted at the very beginning of the match. Moreover, deficient matchmaking algorithms, for example those solely based on skill, are likely to be subject to user manipulation [Caplar et al. 2013]. Given the above reasons, in this section we study the match recommendation problem in MOGs.

Predicting new relationships is in fact a form of the *link prediction problem*, which, given a snapshot of a network, seeks to accurately predict links that will be added to the network in the future [Liben-Nowel and Kleinberg 2003]. In this section, we will assess the performance of traditional link prediction algorithms, with some variations based on our models, for predicting new relationships in MOGs. Combining the tasks of predicting new and repeated relationships, we also propose SAMRA, a Socially Aware Match Recommendation Algorithm that takes social relationships revealed by our model into account. We show that our model not only improves the precision of traditional link prediction approaches, but also, via a domain-specific metric derived from gaming studies [McGonigal 2011], potentially helps players enjoy the game to a higher extent.

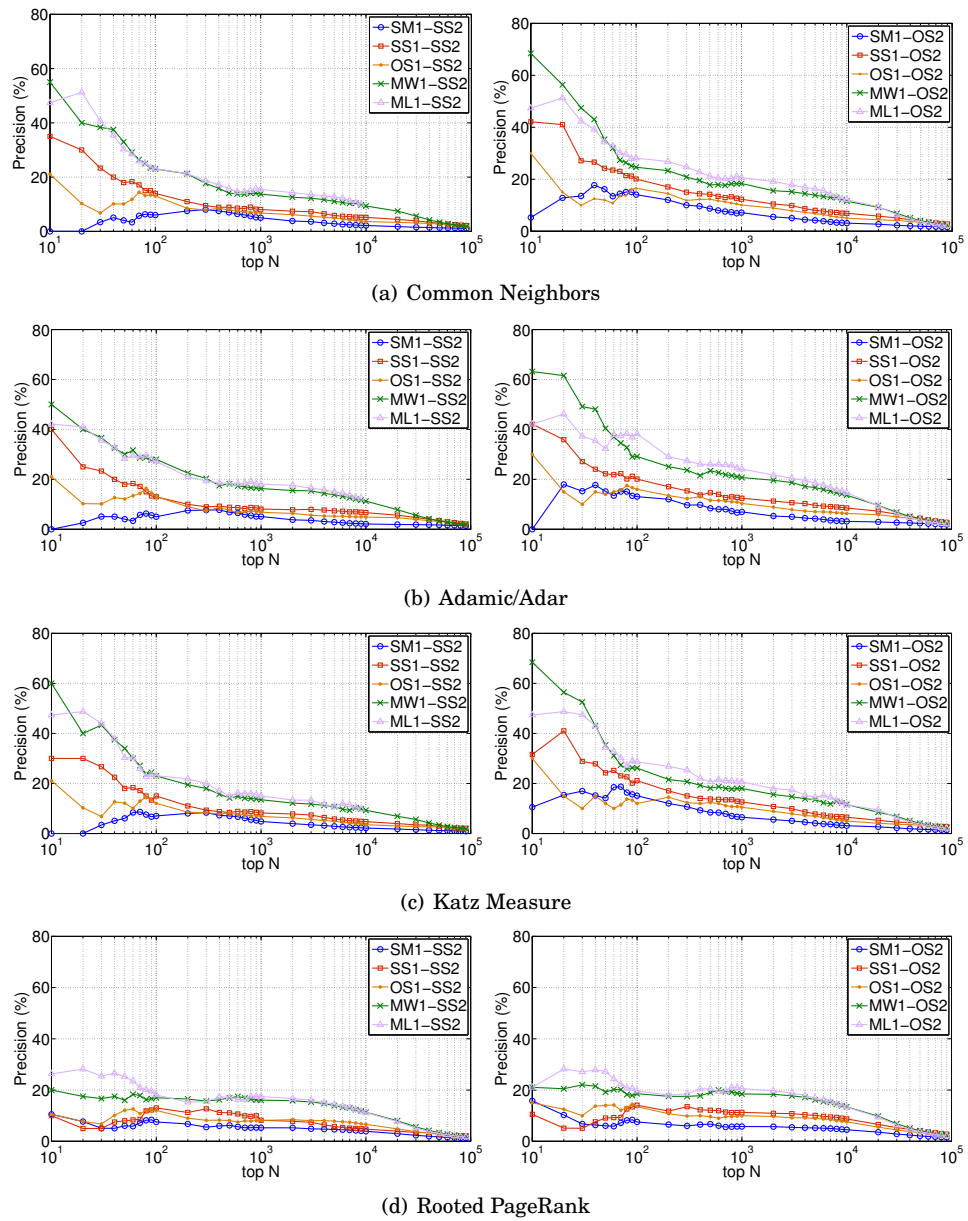


Fig. 12. The performance of link prediction algorithms for match recommendation (horizontal axis with logarithmic scale).

## 7.2. Link prediction approaches applied to MOGs: Predicting new relationships

In this section, we will assess the performance of traditional link prediction algorithms for predicting new relationships in MOGs.

*7.2.1. Link prediction algorithms.* To predict new links, a link prediction algorithm first calculates the similarity between nodes. Assuming that similar nodes are more likely to establish new links, it then produces a list of potential new links. The definition of similarity varies across algorithms. For our analysis, we consider the following four popular algorithms:

**Common Neighbors.** The idea behind this algorithm is that the larger the intersection of the neighbor sets of any two nodes, the larger the chance of future interactions between them [Liben-Nowel and Kleinberg 2003].

**Adamic/Adar.** This algorithm also measures the intersection of neighbor sets of a user pair, but emphasizes a smaller overlap [Adamic and Adar 2001].

**Katz Measure.** The rationale behind this algorithm is that the more paths exist between any two nodes and the shorter these paths, the larger the chance of future interactions between them [Katz 1953].

**Rooted PageRank.** This algorithm captures the probability of random walks starting from two nodes in the graph to meet each other, and uses this probability to quantify the chance of future interactions between them [Song et al. 2009].

*7.2.2. Experiment setup.* As link predictions are often needed within a short time span, we take one year worth of data of Dota-League, from March 2011 to February 2012, as the example to test the performance of link prediction approaches on match recommendation. First, we divide the data into two parts, the training and the testing data. Next, we generate two sets of interaction graphs based on interactions observed in the training and testing data, indicated by SM1 and SM2 (Same Match), SS1 and SS2 (Same Side), etc., respectively. Then, we run a link prediction algorithm on the training data which produces a list of predicted links ranked in decreasing order of prediction confidence. Finally, we take the set  $P_N$  of top- $N$  links from this prediction list and we check whether these links indeed occur in the testing data. Indicating the set of links in the testing data by  $L_2$ , we use *precision* defined as  $|P_N \cap L_2|/|P_N|$  as the metric to measure the performance of top- $N$  link prediction.

We have tested different partitionings of the data, and we have found that using the first half year as the training data and the second half year as the testing data gives the best prediction performance. We use this partitioning for all the following experiments. Under this partitioning, the number of new links in the testing data as compared to the training data for the SM, SS, OS graphs are 66,612, 18,912, and 25,340, respectively.

*7.2.3. Unitary prediction and hybrid prediction.* We call predictions based on the same type of interactions for training and testing *unitary predictions*, and predictions based on different types of interactions *hybrid predictions*. Traditional link predictions are often unitary, for example, using links in SM1 to predict links in SM2. As our model captures different types of interactions, it provides the opportunity for hybrid predictions. For example, *matches won together* (MW) often generate a strong social attachment and players who have won together are very likely to play on the same side (SS) in the future. Thus, it may be beneficial to use links in MW1 to predict links in SS2.

*7.2.4. Results.* Here, we consider five types of relationships, i.e., SM (Same Match), SS (Same Side), OS (Opposite Side), MW (Matches Won together), and ML (Matches Lost together). As discussed in Section 4, we use  $n = 10$  as the default value to generate the interaction graphs from the training and the testing data. We focus on two prediction

tasks, i.e., predicting player pairs that will play on the same (SS) and opposite sides (OS). Note that the precision of predicting the SS and the OS relationships are in fact lower bounds for the precision of predicting the SM relationship. We use both unitary and hybrid predictions for these tasks. In Fig. 12, we show the precisions of these predictions, from which we obtain the following observations.

First, we see that among the four link prediction algorithms we consider, there are no dominant algorithms that outperform the others for all cases. This result is consistent with previous work [Liben-Nowel and Kleinberg 2003] on evaluating the performance of link prediction algorithms. Secondly, we find that in general, for any link prediction algorithm, a higher value of  $N$  leads to a smaller precision, since the predicted links are ranked in decreasing order of prediction confidence. Thirdly, we observe that the predictions based on MW1 and ML1 perform better than the predictions based on SS1 and OS1, which in turn perform better than the predictions based on SM1. We believe the reason is that, as we use the default value  $n = 10$  for generating the these graphs, the relationships presented in MW1 and ML1 are stronger (i.e., less casual) than in SS1 and OS1, which in turn are stronger than in SM1. For the gaming datasets we study, the relationships inferred from strong relationships in the past are more likely to happen in the future.

Further, we find that predictions based on SS1 perform better than predictions based on OS1. We observed earlier in Section 5.6 that for Dota-League, triadic closure is more likely to happen for the SS relationship than for the OS relationship. These results suggest that the SS is less casual than the OS relationship. On the other hand, predictions based on MW1 and ML1 achieve similar performance. One possible reason is that in Dota-League, winning and losing together do not have significantly different impacts on user relationships.

### 7.3. Beyond precision and link prediction

In the previous section we have assessed the performance of link prediction to the match recommendation problem based on the well-known metric of precision. This analysis does not completely solve the match recommendation problem, for two reasons.

First, precision alone is not sufficient for measuring recommendation quality, since the actual new links in the testing data are affected by the current recommendation algorithms. If a player did not play with recommended players in the period of the testing data, it may be because the current system did not introduce them and the player did not know about the possibility of playing with these players, and not because the recommendation is of low quality. Therefore, we need new metrics beyond precision.

Secondly, traditional link prediction algorithms can only predict new relationships. Match recommendation, on the other hand, requires predicting both the new and the repeated relationships. For the latter case, we need new algorithms that go beyond link prediction.

Our approaches to solve the above two problems are discussed in the following two sections.

### 7.4. Bonding score: a pro-social metric for online games

To solve the first problem in Section 7.3, we propose a new metric for measuring match recommendation quality called the *bonding score*, which takes the social components in games into consideration. There are multiple ways to define bonding score. Nevertheless, our goal is not to propose a unique scoring method, but rather to show how to compare and possibly improve match recommendation based on one such socially aware scoring system.



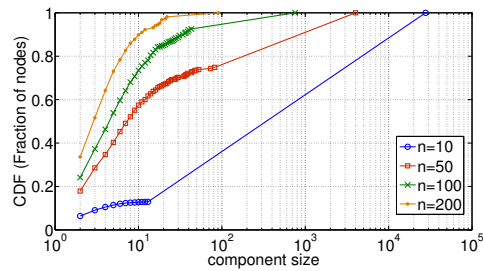


Fig. 13. The CDF of component size for different values of interaction threshold  $n$  (Dota-League, SM graph).

Team 1		Team 2	
Player	Component	Player	Component
a	1	f	2
b	2	g	5
c	1	h	3
d	3	i	6
e	4	j	3

Fig. 14. An example of a match with a bonding score of 7.

To do so, we consider the findings of McGonigal *et al.* [McGonigal 2011], who have pointed out that matches played by players with strong social ties are enjoyed to a higher extent than those played amongst players that have weak or no social ties. In our model, identifying strong social ties amounts to increasing the threshold  $n$  of the number of interactions. As we have shown in Fig. 1, when  $n$  increases, the fraction of nodes in the LCC (Largest Connected Component) decreases dramatically. We further show in Fig. 13 the CDF of the component size in Dota-League’s SM graph, for different values of the threshold  $n$ . We see that, when  $n$  increases, players are grouped into small, intensely interacting components. For example, when  $n = 10$ , less than 10% players are outside the LCC, while when  $n = 100$ , more than 90% players are outside the LCC and form connected components with sizes smaller than 40.

We calculate the bonding score of a match of two teams in the following way. First, we generate the interaction graph for the interaction type we are interested in based on a large value of the threshold  $n$ , so that only strong ties are present in the graph. This interaction graph consists of a number of connected components. Then we calculate the overlap of components across the players of both teams: a match receives one score point for every player (of any of the two teams) who is in a connected component that is represented in the match by at least two players (again, of any of the two teams). As an example, consider the match between the two teams shown in Fig. 14. Here, Components 1, 2, and 3 have more than one player in the match, namely players a and c in Component 1, players b and f in Component 2, and players d, h and j in Component 3, respectively. Therefore, the bonding score of this match is equal to 7.

### 7.5. SAMRA: A Socially Aware Match Recommendation Algorithm

To solve the second problem introduced in Section 7.3, we propose a Socially Aware Match Recommendation Algorithm (SAMRA), which takes the bonding score into consideration.

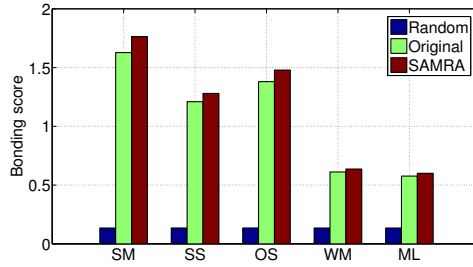


Fig. 15. The average bonding score of all Dota-League matches, for random recommendation, the original recommendation algorithm used in Dota-League, and SAMRA. We use  $n = 10$  to generate the interaction graphs from the training data.

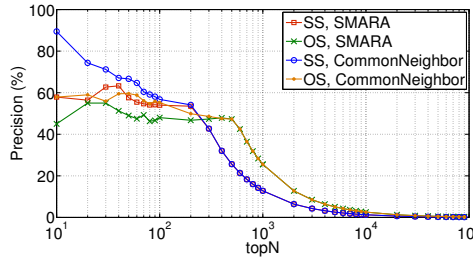


Fig. 16. The precision of SAMRA and Common Neighbors in predicting new relationships (horizontal axis with logarithmic scale). To have a fair comparison, for the Common Neighbor method, we use  $n = 100$  to generate the interaction graphs from the training data.

In SAMRA, first, for each 10-minute time interval<sup>4</sup>, we build a list of all the players who are online and have not yet been assigned to a match. Secondly, for each of these online players, we compute their connected components based on the interaction graphs extracted with a large value for threshold  $n$ . Thirdly, we rank these components in decreasing order of their size, and from the top to the bottom of this ranking, we assign all online players from the same component to the same match if size permits; otherwise we split the component into a number of parts, with the number of players in each part equal to the match size, who then are assigned to the same match.

We test the performance of SAMRA using both the bonding score and the precision. We set  $n = 100$  to generate the interaction graphs based on which our match recommendation algorithm works. The online time of users and the duration of matches are obtained from the original data.

**7.5.1. Performance: bonding score.** First, we compare the bonding scores obtained via SAMRA with those observed in practice. For comparison, we have also scored the matches obtained via random matching. We use the Dota-League dataset as the example. The results are shown in Fig. 15. We see that SAMRA, albeit simplistic, can reach higher match bonding scores than its non-social counterparts. We attribute this improvement to the difficulty of seeing whether friends are online in real systems due to shortcomings in the offered matchmaking methods. We have tested different values of  $n$  and we have obtained similar results.

**7.5.2. Performance: precision.** Here, we test the precision of SAMRA for predicting new relationships and repeated relationships, respectively.

<sup>4</sup>Most matches in Dota-League last for around 40 minutes. Therefore, 10-minute intervals provide enough granularity for capturing most of the online sessions.

**Predicting new relationships:** Using the same training and testing data as in Section 7.2, we calculate the precision of SAMRA in predicting new relationships, i.e., the percentage of player pairs it recommends that indeed occur in the future. We compare SAMRA with traditional link prediction algorithms exemplified by Common Neighbors.

As we use  $n = 100$  to generate the interaction graphs based on which SAMRA works, to have a fair comparison, for the Common Neighbor method, we use  $n = 100$  to generate the interaction graphs from the training data as well. (Recall that in Section 7.2 we used  $n = 10$  as the default value.) We consider the same prediction tasks as in Section 7.2, i.e., predicting the SS and OS relationships, and we focus on unitary predictions. The results are shown in Fig. 16.

We see that Common Neighbors achieve a better top- $N$  precision for a smaller value of  $N$  (i.e.,  $N < 250$ ) while SAMRA performs better for a larger value of  $N$  (i.e.,  $N > 250$ ). As Common Neighbors ranks predicted links based on prediction confidence, when  $N$  increases, the decrease of its precision is more skewed compared to SAMRA.

**Predicting repeated relationships:** As link prediction algorithms are not able to predict repeated relationships, here we only show the results of SAMRA. For the task of predicting repeated SS and OS relationship, we find that SAMRA achieves precisions of 60.60% and 65.88%, respectively.

We have tested different values of  $n$  for generating interaction graphs based on which SAMRA works. Consistent with our intuition, the choice of  $n$  imposes trade-offs on the recommendation performance: a larger value of  $n$  extracts stronger relationships that will help our recommendation algorithm to achieve better precision, but on the other hand, it also excludes players that are without strong relationships and for these players it cannot make any recommendation. Therefore, in general, a larger value of  $n$  yields a higher top- $N$  precision when  $N$  is small and a lower top- $N$  precision when  $N$  is large. Depending on the design philosophy, MOG community administrators could use large values of  $n$  to emphasize the prediction precision or small values of  $n$  to cover most of their users.

## 8. RELATED WORK

In this paper, we propose graph models to analyze social relationships and network evolutions in Online Multiplayer Games (MOGs). We further use match recommendation as the example to show the application of our models to gaming services. We summarize related work within each research topic as follows.

**Graph models in regular online social networks.** An increasing number of social network analyses adopts the complex network approach, i.e., using graphs to represent user relationship. A comprehensive overview of research on complex networks can be found in [Newman 2003]. Some previous work considers only static relationship like friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009], [Mislove et al. 2007]. In contrast, we consider user interactions that are more dynamic. Some previous work like [Wilson et al. 2009], [Viswanath et al. 2009], [Leskovec et al. 2005], [Xiang et al. 2010], and [Liu et al. 2012] has also considered user interactions. However, in their approaches graphs are extracted based on a single, domain-specific, and usually threshold-based rule for mapping interactions to links. De Choudhury *et al.* [Choudhury et al. 2010] have analyzed the influence of thresholds on graph properties. Nevertheless, they have only considered one type of threshold, i.e., the number of interactions (equivalent to  $n$  used in our paper), and one type of interaction, i.e., email exchange. In contrast, in this paper we conduct a sensitivity study of various thresholds and rules based on different types of interactions.

**Graph models in MOGs.** Within MOGs, Kirman *et al.* [Kirman and Lawson 2009] proposes a graph model in which interactions are simply considered as homogeneous

and are mapped to undirected and unweighted links. Merritt *et al.* [Merritt and Clauset 2013] considers the number of interactions when forming the links, but they still consider interactions as homogeneous. Existing work is complemented by considering and comparing friendship and enemy relationships [Szell and S.Thurner 2012], but they still do not differentiate interactions with various types and strengths. In [Ang 2011], differences in graphs extracted based on different types of interactions are studied, but their dataset is rather small, with only 74 users involved, whereas our datasets in this paper cover millions of users. In [Balint et al. 2011] and [Posea et al. 2010] several graph extraction strategies are investigated, but they do not form a formalism as comprehensive as we propose here.

Our previous work [van de Bovenkamp et al. 2013] and [Iosup et al. 2014] propose graph models to analyze social relationships in MOGs, but they lack a thorough study of the impact of mapping functions and thresholds on the characteristics of the resulting graphs.

**Network evolution models.** Most previous works on network evolution only consider static relationships such as friendship [Garg et al. 2009], [Kairam et al. 2012], [Hu and Wang 2009], where the deletion of links are rare and therefore networks are expected to grow. In contrast, we focus on more dynamic relationships, i.e., user interactions. A number of previous works have considered interactions [Viswanath et al. 2009], [Leskovec et al. 2005], [Merritt and Clauset 2013], [Ribeiro et al. 2013], [Caceres and Berger-Wolf 2013], however, they can be arbitrary when defining network evolution and they often consider only one of the two types, network growth and network dynamics. Instead, we consider both of them and we analyze their differences.

Previous work on network evolution that comes closest to our analysis is [Krings et al. 2012], where the authors have analyzed both the network growth and the network dynamics. Nevertheless, they only consider one type of interaction, i.e., phone calls, and they assume a link exists between two users as long as they have interacted before, regardless of their interaction strength. In contrast, we consider different types of interactions, and we use threshold  $n$  (which reflects the interaction strength) to decide whether links should be added between players. We also analyze the influence of  $n$  on the network evolution models, as well as on the predictions made by the models.

**Applications.** Besides understanding the nature of the network, graph models are useful in many applications, including re-generating and predicting the evolution of the network [Garg et al. 2009], [Kairam et al. 2012], [Leskovec et al. 2005], analyzing information diffusion [Liu et al. 2012], and exporting trustworthy user relationships to social-based distributed applications like Reliable Email and SybilGuard [Wilson et al. 2009]. In this paper, we use match recommendation as the example to show the application of our models. We complement our previous work [van de Bovenkamp et al. 2013] and [Iosup et al. 2014] by showing how our models not only improve the traditional link prediction approaches, but also potentially help players enjoy games to a higher extent.

## 9. CONCLUSION

Multiplayer Online Games (MOGs) have attracted hundreds of millions of players world-wide among whom rich social relationships have developed. With traditional complex network approaches and extensive datasets from three MOGs that contain in total years of behavioral history for millions of players and games, in this paper we are able to propose and evaluate a graph model that captures a variety of social relationships in MOGs. Among many interesting observations, we find that MOGs quickly become disconnected when the constraints for relationships to exist increase, that friendship has a positive influence on user interactions, that MOGs can exhibit growth even when their player activity declines, and that members in the early days of

MOGs are often more committed to the communities. Taking match recommendation as the example, we further study the implications of our model on gaming services. We propose SAMRA, a socially aware match recommendation algorithm that takes social relationships into account. Results show that our model not only improves the precision of traditional link prediction approaches, but also potentially helps players enjoy the games to a higher extent.

### Acknowledgement

This work is partially supported by the National Basic Research Program of China under grant No. 2011CB302603, and the Dutch STW/NWO Veni personal grant @ large (#11881).

### REFERENCES

- L.A. Adamic and E. Adar. 2001. Friends and neighbors on the web. *Social Networks* 25 (2001), 211–230.
- C.S. Ang. 2011. Interaction networks and patterns of guild community in massively multiplayer online games. *Social Network Analysis and Mining* 1, 4 (2011), 341–353.
- M. Balint, V. Posea, A. Dimitriu, and A. Iosup. 2011. An Analysis of Social Gaming Networks in Online and Face to Face Bridge Communities. In *ACM Workshop on Large-scale System and Application Performance (LSAP'11)*.
- R. Sulo Caceres and T. Berger-Wolf. 2013. *Temporal Networks*. Springer Berlin Heidelberg, 25–94 pages.
- N. Caplar, M. Suznjevic, and M. Matijasevic. 2013. Analysis of players in-game performance vs rating: Case study of Heroes of Newerth. In *Proceedings of Foundations of Digital Games Conference*.
- D. Cartwright and F. Harary. 1956. Structure balance: A generalization of Heider's theory. *Psychology Review* 63, 5 (1956), 277–293.
- M. De Choudhury, W.A. Mason, J.M. Hofman, and D.J. Watts. 2010. Inferring Relevant Social Networks from Interpersonal Communication. In *Proceeding of the 12th International World Wide Web Conference (WWW'10)*.
- A. Clauset, C.R. Shalizi, and M.E.J. Newman. 2009. Power-law distributions in empirical data. *SIAM Rev.* 51 (2009), 661–703.
- S. Garg, T. Gupta, N. Carlsson, and A. Mahanti. 2009. Evolution of an online social aggregation network: an empirical study. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference (IMC'09)*.
- F. Heider. 1944. Social perception and phenomenal relations. *Psychological Review* 5 (1944), 358–374.
- F. Heider. 1946. Attitudes and cognitive organization. *Journal of Psychology* 21 (1946), 107–112.
- H. Hu and X. Wang. 2009. Evolution of a large online social network. *Physics Letters A* 373, 12-13 (2009), 1105–1110.
- A. Iosup, R. van de Bovenkamp, S. Shen, A.L. Jia, and F. Kuipers. 2014. An Analysis of Implicit Social Networks in Multiplayer Online Games. *IEEE Internet Computing*. Pre-print available online at <http://www.computer.org/csdl/mags/ic/preprint/06756709-abs.html>. (2014).
- J. Jiang, C. Wilson, X. Wang, W. Sha, P. Huang, Y. Dai, and B.Y. Zhao. 2013. Understanding Latent Interactions in Online Social Networks. *ACM Transactions on the Web (TWEB)* 7, 4 (2013), No.18.
- S.R. Kairam, D.J. Wang, and J. Leskovex. 2012. The life and death of online groups: predicting group growth and longevity. In *Proceedings of the 5th ACM international conference on Web search and data mining (WSDM'12)*.
- L. Katz. 1953. A new status index derived from sociometric analysis. *Psychometrika* 18, 1 (1953), 39–43.
- B. Kirman and S. Lawson. 2009. Hardcore classification: Identifying play styles in social games using network analysis. In *Proceedings of the International Conference on Entertainment Computing (ICEC'09)*.
- G. Krings, M. Karsai, S. Bernharsson, V.D. Blondel, and J. Saramaki. 2012. Effects of time window size and placement on the structure of aggregated networks. *EPJ Data Science* 1, 4 (2012), 1–19.
- J. Leskovec, J. Kleinberg, and C. Faloutsos. 2005. Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceeding of the 12th International World Wide Web Conference (WWW'05)*.
- D. Liben-Nowel and J. Kleinberg. 2003. The link prediction problem for social networks. In *Proceeding of the twelfth international conference on Information and knowledge management (CIKM'03)*.

- X. Liu, Q. He, Y. Tian, W. Lee, J. McPherson, and J. Han. 2012. Event-based social networks: linking the online and offline social worlds. In *Proceeding of the 18th ACM SIGKDD international conference on Knowledge Discovery and Data Mining (KDD'12)*.
- J. McGonigal. 2011. *Reality is Broken: Why Games Make us Better and How They Can Change the World*. Jonathan Cape.
- S. Merritt and A. Clauset. 2013. Social Network Dynamics in a Massive Online Game: Network Turnover, Non-densification, and Team Engagement in Halo Reach. In *Proceeding of the 19th ACM SIGKDD international conference on Knowledge Discovery and Data Mining (KDD) workshop on Mining and Learning with Graphs (MLG)*.
- S. Milgram. 1967. The small world problem. *Psych. Today* 2 (1967), 60–67.
- A. Mislove, M. Marcon, K.P. Gummadi, P. Druschel, and B. Bhattacharjee. 2007. Measurement and analysis of online social networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement conference (IMC'07)*.
- M.E.J. Newman. 2003. The structure and function of complex networks. *SIAM Rev.* 45 (2003), 167–256.
- V. Posea, M. Balint, A. Dimitriu, and A. Iosup. 2010. An analysis of the BBO fans online social gaming community. In *Proceedings of the 9th RoEduNet International Conference (RoEduNet'10)*.
- B. Ribeiro, N. Perra, and A. Baronchelli. 2013. Quantifying the effect of temporal resolution on time-varying networks. *Scientific Reports* 3, 3006 (2013), 1–5.
- J.L. Rodgers and W.A. Nicewander. 1988. Thirteen ways to look at the correlation coefficient. *The American Statistician* 42(1):59-66 (1988).
- H.H. Song, T.W. Cho, V. Dave, Y. Zhang, and L. Qiu. 2009. Scalable proximity estimation and link prediction in online social networks. In *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement conference (IMC'09)*.
- M. Szell and S.Thurner. 2012. Social dynamics in a large-scale online game. *Advances in Complex Systems* 15, 6 (2012), 167–256.
- R. van de Bovenkamp, S. Shen, A. Iosup, and F.A. Kuipers. 2013. Understanding and Recommending Play Relationships in Online Social Gaming. In *the 5th International Conference on COMMunication Systems and NETWORKS (COMSNETS'13)*.
- B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. 2009. On the evolution of user interaction in Facebook. In *the Second ACM SIGCOMM Workshop on Online Social Networks (WOSN'09)*.
- D.J. Watts and S.H. Strogatz. 1998. Collective dynamics of small-world networks. *Nature* 393 (1998), 440–442.
- C. Wilson, B. Boe, A. Sala, K. P. N. Puttaswamy, and B. Y. Zhao. 2009. User Interactions in Social Networks and their Implications. In *Proceedings of the 4th ACM European conference on Computer systems (EuroSys'09)*.
- R. Xiang, J. Neville, and M. Rogati. 2010. Modeling relationship strength in online social networks. In *Proceeding of the 19th International World Wide Web Conference (WWW'10)*.