Analyzing Implicit Social Networks in Multiplayer Online Games

Understanding the social structures that people implicitly form when playing networked games helps developers create innovative gaming services to benefit both players and operators. But how can we extract and analyze this implicit social structure? The authors’ proposed formalism suggests various ways to map interactions to social structure. Applying this formalism to real-world data collected from three game genres reveals the implications of the mappings on in-game and gaming-related services, ranging from network and socially aware player matchmaking to an investigation of social network robustness against player departure.
via community websites, which might include services that match players to a game instance, yet aren’t affiliated with the game developer.

In the absence of explicitly expressed relationships, if we want to understand the social networks of current SNGs, we must extract the implicit social structure indicated by regular player activity. However, in contrast to general social networks, no set of meaningful interactions has been defined for SNGs. Moreover, in MOBAs, activities are match- and team-oriented, rather than individual. We address these challenges through a formalism for extracting implicit social structure from a set of SNG-related, meaningful interactions. We extend our previous work\(^1\) by showing that SNGs’ implicit social structure is strong, rather than the result of chance encounters, and that, for MOBAs, the network core (high-degree nodes) is robust over time.

In addition, we apply our formalism to RTS and massively multiplayer online first-person shooter (MMOFPS) games, showing evidence that RTS games exhibit even stronger team structure than MOBAs. This indicates that modern MMOFPSs might require operator-side mechanisms to spur the formation of meaningful social structures. Connecting theory to practice, we also show how the extracted implicit social graphs can help improve game play, retain players and groups, tune the technological platform on which the games operate, and so on.

### SNGs without an Explicit S

Defense of the Ancients (DotA) is an archetypal MOBA game in which social relationships, such as same-clan membership and friendship, can improve the gameplay experience.\(^2\) DotA is a 5-versus-5-player game. Each player controls an in-game avatar, and teams try to conquer the opposite side’s main building. Each game lasts roughly 40 minutes and includes many strategic elements, ranging from team operation to resource micromanagement.

To examine implicit relationships in DotA, we collected data for the DotA communities DotA-League and DotAlicious. Both communities, independently of the game developer, run their own servers, maintain lists of tournaments and results, and publish information such as player rankings. Through these communities’ websites, we obtained all the unique matches and, for each match, the start time, duration, and community identifiers of participating players. After sanitizing the data, we obtained for DotA-League a dataset containing 1,470,786 matches occurring between November 2008 and July 2011; for DotAlicious, the dataset contained 617,069 matches occurring between April 2010 and February 2012.

### Identifying Implicit Social Relationships

Traditionally, online social networks let their users explicitly define their relationships; for example, Facebook users create friendship relationships with other users. In contrast, communities in SNGs are built ad hoc, with tools and services that don’t support explicit relationships or that might remain unused. Identifying the implicit social relationships in SNG communities requires new theoretical tools, such as the formalism we propose here.

### Social Relationships in SNGs

A mapping is a set of rules that define the nodes and links in a graph. Formally, a dataset \(D\) is mapped onto a graph \(G\) via a mapping function \(M(D)\), which maps individual players to nodes (graph vertices) and relationships between players to links (graph edges).

Instead of proposing a graph model, we focus on formalizing mappings that extract graphs from real data. Because many social network metrics apply only to unweighted graphs, relations are often considered as links only if their weight exceeds a threshold. Thresholding, therefore, has an important impact on the resulting graph.

Related to our work, interaction graphs map users of social applications to nodes,\(^3\) and events involving user pairs to links, via a threshold-based rule (we discuss related work throughout this article and elsewhere\(^1\)).

### Interaction Graphs in MOBAs

A mapping is meaningful if it leads to distinct yet reasonable views of implicit social networks appearing in networked games. We identify six types of player-to-player interactions:

- **SM** — two players are present in the *same match*.
- **SS** — two players are present on the *same side* of a match.
- **OS** — two players are present on opposing sides of a match.
- **MW** — two players *won* a match together.
- **ML** — two players *lost* a match together.
- **PP** — *(directed link in the social relationship graph)* for a player, when present in at least \(x\) percent of another player’s matches \((x = 10\%\) in this article). This interaction is effec-
Networked Games

tively PP(SM). Similarly, we can define PP(SS), and so on.

To extract the social networks that correspond to various relationship types, we extract a graph for each mapping using a threshold \( n \), which reflects the minimum number of events that must have occurred between two users for a relationship to exist. For example, for SM\( (n = 2) \), a link exists between a pair of players if and only if they were both present in at least two matches in the input dataset. A second threshold, \( t \), which limits the duration of effect for any interaction, is less relevant, as we explain later.

The set of mappings we propose here isn’t exhaustive. For instance, this formalism can support more complex mappings, such as “played against each other at least 10 times, connected through ADSL2, while located in the same country.” Moreover, the interactions in the set aren’t independent. For example, we can view the SS mapping as a specialization of the SM mapping.

Application to Exemplary MOBAs

In this article, we focus on three methodological questions.

**Question 1.** Are the relationships we identify the result of players being simultaneously online by chance? To answer this question, we first create a reference model by randomizing, for any window of length \( w \) minutes, the interactions observed in the MOBA datasets. We randomize, for example, the SM mapping by taking the players from all matches that started within the current time window and randomly assigning them to matches. Because the SM mapping doesn’t consider team information, the match assignment comes down to forming random groups of 10 players from the entire list who were active in the time window. A single player can be on the list multiple times, and the random groups must have 10 different players.

We ran the parameter \( w \) from 1 to \( \infty \) and depict the results, together with the original data, in Figure 1a. Whereas the results for \( w = 1 \) leave little room for randomization, the results for \( w = \infty \) randomize the entire dataset. In Figure 1a, the curves for \( w = 1 \) and the original data have a power-law-like shape. The curves for various values of \( w \) follow the \( w = 1 \) (original) curve for link weights of up to about 15 matches played together, but afterward take an exponential-like shape, which indicates that they are more likely to be the result of chance than of intended user behavior. Curves are markedly different for small time windows, showing that players are unlikely to play together often just because they happen to be online simultaneously.

The results for the other game genres (introduced later), which Figures 1b and 1c depict, show similar yet not so pronounced behavior. Although players don’t play together nearly as often in other genres’ datasets as in the MOBA datasets, randomization within only small time windows lowers the link weights.

We conclude that it is unlikely that the relationships we identify are due to chance encounters among players and, instead, indicate conscious, possibly out-of-game agreements between players.

**Question 2.** Do players (nodes) preserve their high-degree property over time? If so, the networks these players form could be robust against natural degradation, with implications for the long-term retention of the most active players.

For each MOBA community, we first divide its last year’s gaming relationships into two parts: the first half year is training data, and the second half year is testing data. We use only players who appear in both datasets — about 60 percent of the training-data players. For different degrees of players in the training dataset, Figure 1d plots the average number of links they form in the testing dataset. From the high value and positive correlation coefficient (0.6233 for DotA-League), we determine that players with higher degrees in the training dataset robustly establish more new links in the testing dataset than other players.

**Question 3.** Are the mappings we propose meaningful for MOBAs? To answer this question, we extract the interaction graphs and compute a variety of graph metrics for each of our mappings. Table 1 summarizes the results, which lead to the following conclusions.

First, side-specific interactions (SS and OS) are meaningful. For example, players are more likely to be on opposing sides (OS) than on the same side (SS) in DotA-League (for example, higher \( N \) and \( L \) in Table 1); for DotAlicious, the reverse is true. Game designers could enable OS links by letting players explicitly identify their foes.

Second, outcome-specific interactions (MW and ML) are meaningful. For example, for DotAlicious only, MW leads to the formation of more
relationships. Game operators could exploit this in matchmaking services.

Finally, relative joint-participation (PP) is meaningful. For example, for PP(SM), the number of nodes in the graph decreases quickly with the \( n \) threshold. Identifying the players who play almost exclusively together can be critical to player retention.

**Application to Other Game Genres**

Among the most popular genres today, RTS games ask players to balance strategic and tactical decisions, often every second, while competing with other players for resources. Although faster-paced, MMOFPS games test players’ tactical teamwork when disputing a territory with other teams. We could expect RTS and MMOFPS games to lead to similar interaction graphs as MOBAs: naturally emerging social structures centered around highly active players. However, these game genres have different match scales and team-versus-team balance than MOBAs. Moreover, RTS games can stimulate individualistic gameplay, whereas MMOFPS games might have teams that are too large to be robust.

We collected then analyzed two additional datasets (Table 2): for the RTS game StarCraft II (SC2) from March 2012 to August 2013; and for the MMOFPS game World of Tanks (WoT) from August 2010 to July 2013. For each of these popular games, we collected more than 75,000

---

**Figure 1.** Results for methodological questions 1 (a–c) and 2 (d). (a) Frequency of link weight occurrence in the reference model for five different intervals and the original data for Defense of the Ancients (DotA). The mapping is SM (same match). (b) Frequency of link weight occurrence in the reference model for five different intervals and the original data for StarCraft II (SC2). The mapping is SM. (c) Frequency of link weight occurrence in the reference model for five different intervals and the original data for World of Tanks. The mapping is SM. (d) Continued activity for gaming relationships in DotA-League (SM with \( n = 10 \)).
matches, played by more than 80,000 SC2 and more than 900,000 WoT players. SC2 matches aren’t generally played in equally sized teams, and 92 percent of our dataset’s matches are 1-versus-1-player. In contrast, 98 percent of WoT matches are 15-versus-15-player, but such large teams can be much harder to maintain over time than those found in typical MOBAs, due to inevitable player churn.

For SC2, the mappings lead to small graphs, with many small, connected components. Most players participate in 1-versus-1-player matches, but the 8 percent of players who do play in larger groups tend to play against each other more than together (N = 611 for the OS mapping, versus 314 for SS). When players do play on the same side, winning tends to strengthen the teams (N = 212 for the MW mapping, versus 95 for ML), just as we saw for the DotAlicious dataset. The connected components are strongly connected, yet small. The connected components of the mappings that extract same-team graphs are highly clustered, whereas the largest component for the OS mapping is a tree. The clustering coefficients observed in the various RTS networks indicate much stronger team relationships in RTS games than in MOBAs. Because RTS games haven’t shown a trend of greatly increasing the number of players in the same instance over the past decade, we hypothesize that RTS games will continue to spawn tightly coupled teams that always play together. Such teams are naturally vulnerable to player departures.

For WoT, the large team size makes organizing teams difficult: the biggest connected components for all mappings are not large. Similarly to SC2 and DotAlicious, in WoT, the players who do play together often do so on the same team and, again, players who play together are more likely to win than lose. Given that modern FPS games are played in increasingly larger teams, with 32-versus-32-player games not uncommon, we conclude that MMOFPS games will require additional mechanisms if they are to develop a robust social structure. Moreover, even more so than in the SC2 datasets, many players play only one or a few games: 69 percent of the more than 900,000 players played only once or twice. This is another area in which developers could use the emerging social structures among their players to increase the number of players that continue playing the game.

We conclude that we can apply our formalism to other game genres, where it could lead to new findings versus MOBAs. We suggest that even communities of popular networked games could benefit from new mechanisms that foster denser interaction graphs.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>DotA-League</th>
<th>DotAlicious</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>31,834</td>
<td>26,373</td>
</tr>
<tr>
<td>Nlc</td>
<td>27,720</td>
<td>18,047</td>
</tr>
<tr>
<td>L</td>
<td>202,576</td>
<td>30,680</td>
</tr>
<tr>
<td>Llc</td>
<td>199,316</td>
<td>17,686</td>
</tr>
<tr>
<td>d</td>
<td>4.00</td>
<td>2.46</td>
</tr>
<tr>
<td>d lc</td>
<td>5.19</td>
<td>4.10</td>
</tr>
<tr>
<td>µ</td>
<td>0.0301</td>
<td>0.0060</td>
</tr>
<tr>
<td>h</td>
<td>4.42</td>
<td>6.30</td>
</tr>
<tr>
<td>D</td>
<td>14</td>
<td>28</td>
</tr>
<tr>
<td>C</td>
<td>0.37</td>
<td>0.41</td>
</tr>
<tr>
<td>ρ</td>
<td>0.13</td>
<td>0.25</td>
</tr>
<tr>
<td>Bm</td>
<td>0.04</td>
<td>0.09</td>
</tr>
<tr>
<td>Cm</td>
<td>85</td>
<td>41</td>
</tr>
</tbody>
</table>

*Metrics for n = 10. We present the number of nodes N, number of nodes in largest connected component Nlc, number of links L, number of links in largest connected component Llc, link density d, link density of largest connected component d lc, algebraic connectivity µ, average hop count h, diameter D, average clustering coefficient C, assortativity ρ, maximum betweenness Bm, and maximum coreness Cm.*
Analyzing Implicit Social Networks in Multiplayer Online Games

**Application to SNG Services**

How can game operators leverage social networking elements to improve gaming services? Let’s look at two exemplary answers.

**Socially and Network-Aware Matchmaking**

Matching players at the start of a game can significantly affect the gameplay experience. Game operators already deploy gaming services that perform matchmaking while considering network latency. In contrast, a socially aware matchmaking service assigns players to matches, trying to ensure that those in the same social, rather than latency-based, cluster play together. We revisit the example of a socially aware matchmaking service1 by also considering network latency.

The result of a socially aware algorithm. First, for each sliding window ($\tau = 10$ minute interval), the algorithm builds a list of all the players who are online. Second, from the social graph, the algorithm computes each player’s cluster membership. Third, from the largest online players’ cluster to the smallest, the algorithm assigns all online players from the same cluster to new matches if size permits; otherwise, it divides the cluster into two parts, and assigns players from one part into new scheduled matches.

Figure 2a sketches computing the score for an exemplary match. Team 1 consists of players A through E (in the “Player” column); team 2 consists of players F through J. The “Cluster” column records each player’s cluster identifier. A match receives one point for every same-cluster player present when at least two same-cluster players are present. In this example, players A and C (cluster 1) and players B and F (cluster 2) receive two points, whereas players D, H, and J (cluster 3) receive three points. Players E, G, and I have no fellow cluster members in the match and receive zero points. In total, this match receives seven points. To favor small clusters, which can lead to novel human emotions,4 our scoring system doesn’t consider the largest cluster when assigning points.

We compare our matchmaking algorithm with those observed in practice in MOBAs, as regards average scores (utility). Figure 2b shows selected results.

Expectedly, random matchmaking, which many gaming communities still use, leads to very low utility. Surprisingly, our simple socially aware matchmaking algorithm also exceeds the performance of the matchmaking algorithm DotAlicious’s operators use. This is because the limited community tools available in practice don’t make all players aware that some of their friends are online and thus let them join other, lower-utility matches.

**The barrier of network characteristics.** Using the geographical location gleaned from MOBA datasets, we estimate possible latency conflicts — for

---

**Table 2. Mapping results for real-time strategy and massively multiplayer online first-person shooter games.**

<table>
<thead>
<tr>
<th>Metrics</th>
<th>StarCraft II (RTS)</th>
<th>World of Tanks (MMOFPS)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SM</td>
<td>OS</td>
</tr>
<tr>
<td>$N$</td>
<td>907</td>
<td>611</td>
</tr>
<tr>
<td>$N_{lc}$</td>
<td>31</td>
<td>22</td>
</tr>
<tr>
<td>$L$</td>
<td>748</td>
<td>404</td>
</tr>
<tr>
<td>$L_{lc}$</td>
<td>58</td>
<td>21</td>
</tr>
<tr>
<td>$d$ ($\times 10^{-4}$)</td>
<td>18</td>
<td>22</td>
</tr>
<tr>
<td>$d_{lc}$ ($\times 10^{-4}$)</td>
<td>1.247</td>
<td>909.10</td>
</tr>
<tr>
<td>$D$</td>
<td>2</td>
<td>8</td>
</tr>
<tr>
<td>$\mathcal{Z}$</td>
<td>0.58</td>
<td>0</td>
</tr>
<tr>
<td>$\rho$</td>
<td>$-0.46$</td>
<td>$-0.45$</td>
</tr>
<tr>
<td>$B_m$</td>
<td>0.91</td>
<td>0.74</td>
</tr>
<tr>
<td>$C_m$</td>
<td>53</td>
<td>11</td>
</tr>
</tbody>
</table>

---

1. Metrics$^1$ for $n = 10$. We present the number of nodes $N$, number of nodes in largest connected component $N_{lc}$, number of links $L$, number of links in largest connected component $L_{lc}$, link density $d$, link density of largest connected component $d_{lc}$, diameter $D$, average clustering coefficient $\mathcal{Z}$, assortativity $\rho$, maximum betweenness $B_m$, and maximum coreness $C_m$. 

---

May/June 2014 7
example, same-match players located in Germany and Asia. We analyze the network latency's impact on our matchmaking algorithm's score and depict the resulting score in Figure 2b. In this scenario, a significant part of the matchmaking score is lost owing to recommendations not considering network latency (yet our matchmaking algorithm still outperforms the original matchmaking). We conclude that combining social and network awareness is important for networked gaming services.

Assessing Social Network Robustness

Because social relationships are important in player retention, the social structure's strength might indicate a community's survival chances. If the network starts breaking down, people might lose interest in the game and stop playing. Operators must assess both the strengths and weaknesses in their games' social structure to be able to stimulate growth or prevent a collapse. Conversely, competitors could try to lure away key players (hubs), who could in turn sway others.

The anatomy of an attack. To assess social network robustness, we conducted a threshold-based degree attack on it. For each mapping, we iteratively removed the top-K players, according to their degrees in the extracted graph, in decreasing order. Removing players removes either their matches (match attack) or their entire connected component (hubS attack). Then, we reapplied the mapping to the remaining matches to get a new network, and output this new network's size and largest component. We performed match and hub attacks on DotAlicious and DotA-League; Figure 3 depicts selected results. We didn't conduct experiments in which players form new clans (network rewiring), which represents the opposite of our scenario. In our experience as gamers, when a member of a strongly connected group leaves (for another game), the whole group departs as well.

The aftermath of an attack. We find that both match and hub attacks on MOBAs are very efficient. For match attacks (Figure 3a), removing the top 1,000 players (only 1.5 percent) can reduce the network's size by 15–60 percent of its initial size, and the largest component's size to less than 10 players. For hub attacks (Figure 3b), removing only the top 100 players can cause the network to implode. A social network collapse also implies the collapse of network traffic, which can waste pre-provisioned networked resources.

Understanding the social relationships between players can help game operators improve social network robustness by identifying and motivating the key players. Our formalism provides important identification tools, but motivating players remains an open problem.
The field of social network research applied to networked games is rich and could lead to important improvements in gameplay, with direct repercussions to networked-resource consumption and quality of experience. We identify several challenges and opportunities related to our study.

Figure 3. Results of match and hub attacks on the social network for $n = 28$ and $K \in [1, 1000]$.
(a) Effect of lost players on network size during a match attack for DotAlicious. On the left, we can see the size of the remaining network, whereas the right shows the size of the largest connected component in the remaining network. (b) Effect of lost players on match count during a hub attack for (left) DotAlicious and (right) DotA-League. (c) Zoom-in of middle plot for $K \in [1, 10]$ removed players.
First, we could expand the mappings set to provide a richer framework for implicit relationships. The framework could focus on temporal aspects such as loose (dense) interactions over long (short) time periods.

We could also complement our work with theories from other disciplines. The prosocial emotions appearing in games could have important implications, so exploring them would be beneficial. From our datasets, which are now publicly available through the Game Trace Archive (http://gta.st.ewi.tudelft.nl/), researchers could infer finer-grained relationships from the combination of explicit friendship relationships and implicit interaction graphs, and test them against theories developed for complex networks, sociology, and psychology.

Finally, we could apply our formalism to networked game services. Our work’s main purpose is to support (future) social game services. We anticipate use in services such as player management and retention through matchmaking recommendations and identification of key players, and the design and tuning of capacity planning and management systems through prediction of graph evolution.

Acknowledgments
We thank the gaming communities for making their data available online. The datasets are available through the Game Trace Archive (see Y. Guo and A. Iosup, “The Game Trace Archive,” ACM NETGAMES, 2012). This research was partly supported by the EU FP7 Network of Excellence in Internet Science EINS (project number 288021), the STW/NWO Veni grant @larGe (11881), a joint China Scholarship Council and Delft University of Technology grant, and by the National Basic Research Program of China (973) under grant number 2011CB302603.

References

Alexandru Iosup is an assistant professor with the Parallel and Distributed Systems (PDS) group at Delft University of Technology (TU Delft). His research interests include large-scale distributed systems and their applications. Iosup received a PhD in computer science from TU Delft. He is a member of IEEE, ACM, and the Standard Performance Evaluation Corporation (SPEC), and chair of the SPEC Cloud Working Group. Contact him at a.iosup@tudelft.nl.

Ruud van de Bovenkamp is a PhD student in the Network Architectures and Services (NAS) group at TU Delft. His research focuses on complex networks, including those forming in multiplayer online games. Van de Bovenkamp received an MSc in telecommunications from TU Delft and an MA (cum laude) in Old English literature from Universiteit Leiden. Contact him at r.vandebovenkamp@tudelft.nl.

Siqi Shen is a PhD student with the PDS group at TU Delft. His research focuses on massivizing online games, including via social networks forming in multiplayer online games to better provision and allocate servers and other resources. Shen received an MSc in computer engineering from the National University of Defense Technology, China. Contact him at s.shen@tudelft.nl.

Adele Lu Jia is a post-doctoral researcher with the PDS group at TU Delft. Her research focuses on the social aspects of large-scale distributed systems, applied to peer-to-peer data-sharing networks. Lu Jia received a PhD in communication engineering from TU Delft. Contact her at l.jia@tudelft.nl.

Fernando Kuipers is an associate professor with the NAS group at TU Delft. His research interests include routing and network algorithms, complex networks, and quality of service and experience. Kuipers received a PhD (cum laude) in electrical engineering from TU Delft. He is a member of the executive committee of the IEEE Benelux chapter on communications and vehicular technology, and a senior member of IEEE. Contact him at f.a.kuipers@tudelft.nl.