International Conference on Internet Studies,
July 18-19, 2015, Tokyo, Japan

LOM: A LEADER ORIENTED MATCHMAKING ALGORITHM FOR MULTIPLAYER ONLINE GAMES

Jehn-Ruey Jiang,
National Central University, Taiwan
jrjiang@csie.ncu.edu.tw

Guan-Yi Sung,
National Central University, Taiwan
ff_f146@yahoo.com.tw

Jih-Wei Wu,
National Central University, Taiwan
jihwei.wu@msa.hinet.net

ABSTRACT

In multiplayer online games (MOGs), matchmaking is the process that groups players into online game sessions. There are generally two types of matchmaking criteria: connection-based and skill-based. By the connection-based criterion, players with higher mutual network connection speeds are grouped together. By the skill-based criterion, players with close skill ratings are grouped together. In this paper, we propose a new criterion, association-based criterion, by which players with high association are grouped together. According to the association-based criterion, we also propose a new matchmaking algorithm, called LOM (Leader Oriented Matchmaking), using the concept of the minimum-cost maximum-flow algorithm to group players into game sessions in an optimized way. We perform simulations for LOM and other related algorithms for the sake of comparisons. We observe that although LOM has the longest execution time, it has the best average association per session.

Keywords: Multiplayer online games, matchmaking, minimum cost maximum flow algorithm

---

1. INTRODUCTION

Multiplayer Online Games (MOGs) is a popular networked game genre for the last decades. Geographically distributed players can join the same game and interact with each other simultaneously. Typical examples of MOGs are the Call of Duty [17], a first person shooter (FPS) game, the Dragon Age [18], a role playing game (RPG), and the World of Warcraft [19], a real-time strategy (RTS) game. Online gaming is currently a rapidly growing industry, and game-developing companies are making games appealing to a wide audience to increase revenues.

For either client/server (C/S) or peer-to-peer (P2P) MOGs [2][9][8], matchmaking is closely related to game player satisfaction and enjoyment, which in turn affect the game's appeal. A matchmaking algorithm groups a specific number of players into a team, and groups a specific number of teams into a game session or tournament based on desired matchmaking criteria. Note that there may be only one session or multiple independent parallel sessions running at the same time. Also note that some MOGs have no notion of teams; in such games, players are grouped into sessions directly. Players are impatient and not willing to spend much time in the matchmaking place (e.g., a game portal) waiting for a game session or tournament to start. A matchmaking algorithm hence should be fast. Moreover, players prefer balanced matchmaking, in which each of the participating teams has a fair chance of winning. A matchmaking algorithm hence should also be balanced.

In general, matchmaking algorithms can be classified as connection-based and skill-based. Connection-based matchmaking pursues balanced network connection quality to gain better game playing experiences, so P2P MOG players with similar mutual network connection speeds are matched up and C/S MOG players with similar server connection speeds are matched up. In skill-based matchmaking, players are estimated with a skill rating system on the basis of their game performances and experiences [3][7][16], and players with close skill ratings are matched up.

In this paper, we propose a new matchmaking criterion, namely the **association-based criterion**, by which players with high association are grouped into a session in a game that can accommodate a large enough number of independently running parallel sessions. According to the association-based criterion, a player has association degrees, each of which corresponds to the weight (or degree) of the association between the player and one of preselected session leaders. A player is added into a session whose leader has the highest association degree with the player. We design a matchmaking algorithm, called **Leader Oriented Matchmaking (LOM)**, to realize matchmaking according to the association-based criterion on the basis of the minimum-cost maximum-flow (MCMF) algorithm [5]. LOM is fast and balanced in the sense that its time complexity is polynomial and it has the best average association per session.

The rest of this paper is organized as follows. In Section 2, we review some related work. We explain the association-based criterion and elaborate the proposed LOM algorithm in Section 3. In Section 4, we evaluate LOM and other related matchmaking algorithms for the sake of comparisons. Finally we conclude the paper with Section 5.

2. RELATED WORK

In the initial stage of MOGs development, players were asked to select some specific servers according to some conditions, such as their regions, network connection states and server loads, to join the games. Selecting a game server is equivalent to selecting a game space and players in the space as possible playmates to start a game session. Since most players used PCs as their main gaming devices in the early stage, manually selecting a game server was a simple and feasible way to match up players. However, current gaming apparatus and networking technologies have been highly developed into a new stage having novel devices, such as smart phones, mobile networks and somatosensory gaming systems. Hence, the traditional server selecting and player matchmaking procedure is not intuitive and too complicated for game players to do. Automatic matchmaking (or simply matchmaking) was thus emerged. With the diversification of gaming devices and the popularity of online games, matchmaking received more and more attention because of its intuitive and easy operability. Many PC-based MOGs even adopt matchmaking as their default player arrangement mechanism. Matchmaking also attracted a lot of attention of academic research [1][4][6][10][11][12][13][14][15]. Below, we describe some research most related to our proposed algorithm.

Agarwal and Lorch [1] designed a connection based matchmaking algorithm, named Htrae, for P2P MOG. Htrae is a geographical based network latency prediction system for estimating the latency between two machines on Internet to cluster players so that those in the same game session have low latency to each other. Htrae synthesizes geolocations for all machines with a network coordination system, so it can estimate latency between two machines quickly and accurately.

Manweiler et al. [13] proposed a connection based matchmaking system, called Switchboard, to efficiently group players into game sessions for MOGs on cellular networks. They studied how the cellular network latencies affect the performance of MOGs and considered that a game matchmaking service needs to know the cellular network latency between game players and quickly group players into viable game sessions. They perform experiments to investigate four strategies about placing servers (i.e., P2P, using a single server, using two servers, and using many geo-distributed servers) for Switchboard performance evaluation. The experiments show that Switchboard achieves scalability both in measurement overheads and computation overheads.

The main concept of skill-based matchmaking is to match up players of similar skill levels. On the one hand, skill rating or ranking is difficult in MOGs; player skill levels are hard to obtain and predict. On the other hand, as mentioned by Delalleau et al. in [4], considering only skill levels is not enough for designing good matchmaking systems. Delalleau et al. therefore proposed an advanced matchmaking system to improve matchmaking results by collecting richer player profiles and player statistics within the game. They argued that fun is most important goal of games, and tried to use fun as the main criterion in the matchmaking system. The *FunNet* model which is constructed by the neural network is used to find out the significant factors that determine the "fun score". They used *FunNet* to carry out an experiment with *Ghost Recon Online* and set up an in-game survey to gather player feedbacks about their gaming experience. The experiment results show that the more proper "fun factors" are, the higher the "overall fun score" is.

Véron et al. [15] pointed out that many features of players, such as habits, behaviors and expectations, are required for designing and implementing a high quality matchmaking service for MOGs. However, these features are not easy to acquire, which causes a certain gap between matchmaking services of games and players' experience. Véron et al. gathered and analyzed more than 28 million game sessions of data from a famous online game, League of Legends, and built a reusable database for establishing effective matchmaking criteria. The authors strongly believe that a database of players' statistics and behaviors can help design future software solutions for gaming.

Laufer et al. [11] considered recommendation systems for game matchmaking in which off-policy policy evaluation is important but standard offline methods can hardly be applied. Their purpose is to build well-balanced matchmaking; however, it is difficult because available training data comes from a policy that is not known perfectly and that is not stochastic. Obtaining data from off-policy dataset by evaluating the reward function is more feasible but is more biased. Laufer et al. thus presented a calibration procedure, namely Stacked Calibration, which is similar to the stacked regression, for removing most biases. Through their simulations, they verified that their Stacked Calibration performs as well as or better than standard offline methods.

Another point of view for matchmaking was proposed by Lanzi et al. [12] focusing on generating a well-designed game maps or game scenarios. The authors found that a good game content design leads to a balanced experience of game. They claimed that the design of the game content has a large impact on the match balancing and that the procedural content might be a promising approach to improve it. The authors used Cube 2: Saubertan, an open source first person shooter online game, to test the correctness of the evolving maps for match balancing in first person shooters.

3. LOM ALGORITHM

This section introduces the proposed LOM (Leader Oriented Matchmaking) algorithm, a fast and balanced matchmaking algorithm for grouping players directly into independently running parallel sessions. LOM needs to select k leaders or representatives of sessions and matches up players on the basis of the association-base matchmaking criterion, which takes the association weight for deciding which game session a player should belong to. Note that each player has k association weights, each of which corresponds to one of the k leaders. As will be shown, LOM uses the MCMF algorithm [5] that takes the association weights as the input for matching up players.

We assume that there are n players in the game portal waiting for matchmaking, and each session should have h players. We also assume that sessions can independently run at the same time; that is, all players in a session can interact with each other for playing games, but players in a session are independent of players in the other session. Therefore, we only need to consider all players in a session for the evaluation of the balance of matchmaking. Initially, LOM randomly selects k session leaders, where $k=\lceil n/h \rceil$. It is noted that there are still other ways to select leaders. For example, LOM may select the first k players waiting in the game portal as leaders.

As shown in Fig. 1, we use a bipartite graph $G=(L, M, E)$ to represent the association relationship between k leaders and n-k members, where $L=\{l_1, l_2, ..., l_k\}$ is the leader set and $M=\{m_1, m_2, ..., m_{n-k}\}$is the member set. To be more precise, there is an edge between a leader l_x in L and a member m_y in M with the weight Wl_xm_y, which corresponds to the association weight between l_x and m_y. We may use many ways to calculate the association weight between two players (nodes). How to precisely calculate the association weight is out of the scope of this paper. We just show a simple example of the calculation as follows. We can keep a profile for every player to record the player properties, such as the habit, interest, country, language, skill level, network condition, and then use player profiles to calculate the association weight between two players.

The closers their properties are, the smaller the association weight of them is. Note that smaller association weights imply larger association degrees.

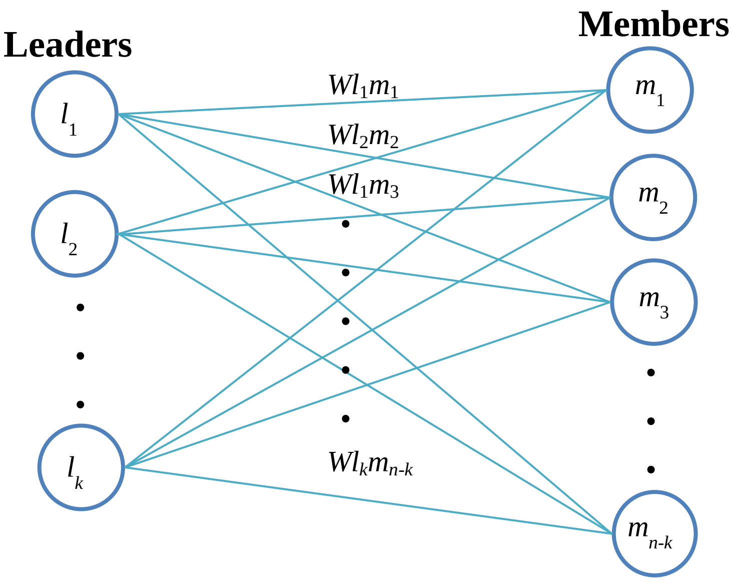

Figure 1. The bipartite graph used by the LOM algorithm

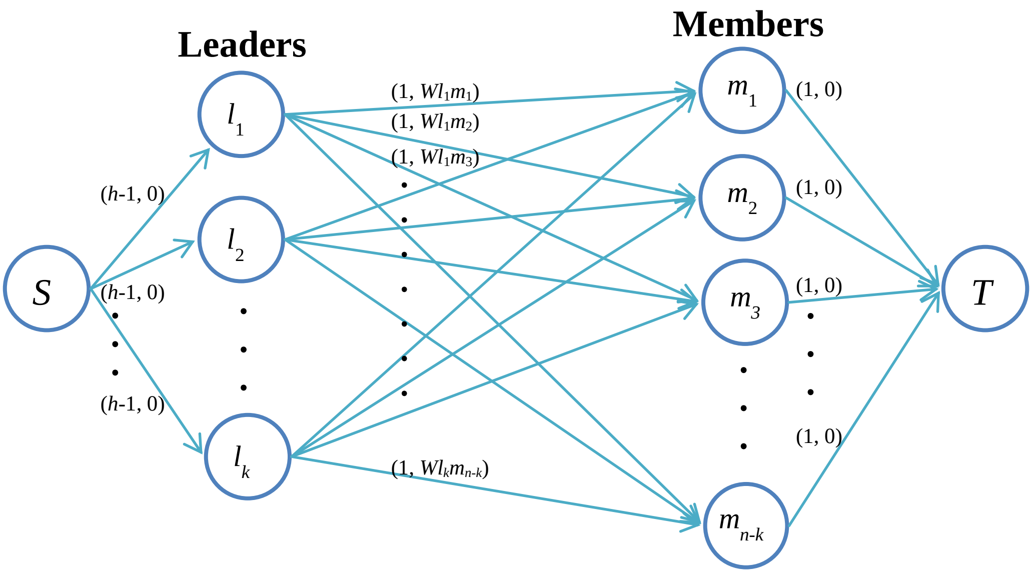

Figure 2. The flow network used by the LOM algorithm

LOM performs matchmaking by matching every leader with exactly h-1 members so that the total association weight (resp., degree) between the leader and all members is minimized (resp., maximized). To be more precise, LOM picks h-1 edges for every leader for the purpose of minimizing the total weight (i.e., maximizing the total association degree) of the picked edges. LOM reaches the goal by transforming the bipartite graph G=(L, M, E) into the flow network, as shown in Fig. 2 and then executing the MCMF algorithm on the flow network.

A flow network is a directed graph where each edge has a capacity and a weight, denoted by the pair (capacity, weight), with two special nodes, the source node *S* and the target node *T*. The source node *S* has only outgoing edges; the target node *T* has only incoming edges; other nodes have both outgoing edges and incoming edges. Edges receive flows, and the number of flows on an edge cannot exceed the capacity of the edge, and the number of flows into a node should equal the number of flows out of it, unless it is the source node or the target node. An outgoing edge of *S* has the capacity *h*-1, so there is *h*-1 flows on every outgoing edge of *S* according to the maximum flow criterion of the MCMF algorithm. Every edge between a leader node and a member node has the capacity 1. To evacuate all the flows coming from the source node, every leader node has to choose *h*-1 outgoing edges, each of which is to receive a flow. Since an outgoing edge of the source node and an incoming edge of the target node have the weight of 0, according to the minimum cost criterion of the MCMF algorithm, all the picked edges have a minimum summation of total weights (i.e., property differences) and hence a maximum summation of association degree.

4. PERFORMANCE EVALUATION

In this section, we evaluate the performance of LOM and other algorithms, namely the M2L Greedy algorithm (denoted by M2L), the L2M Greedy algorithm (denoted by L2M), and the Random algorithm, for the sake of comparison. In the M2L algorithm, every member node, from the first to the last, selects a un-fully-matched leader node with the minimum association weight. Note that a leader node is fully-matched if it has been selected by *h*-1 member nodes; otherwise, it is un-fully-matched. And in the L2M algorithm, every leader node, from the first the last, selects *h*-1unselected member nodes with the top *h*-1 minimum association weights. As to the random algorithm, it just randomly selects *h*-1 edges between a leader node and member nodes. It is merely a reference algorithm for comparisons.

The other parameters of the evaluation are configured as follows: The session size *h* is 10 (i.e., a session has exactly 10 players), and the total number of players waiting in the game portal may be 100, 200, 300, 400 or 500. Note that we assume the system can accommodate a large enough number of sessions to allow all players to play at the same time. The evaluation results of the average association weight of a session are shown in Fig. 3. Note again that smaller association weights imply higher association degrees. By Fig. 3, we can observe that LOM is the best, since it is a globally optimal algorithm to produce the smallest total association weight and hence the smallest average association weight per session. Both L2M and M2L are greedy based algorithms, so their results could be affected by the order of the matchmaking and are worse than those of LOM. However, L2M and M2L have similar average association weights since they employ similar greedy heuristic. As expected, the random algorithm has the worst results.

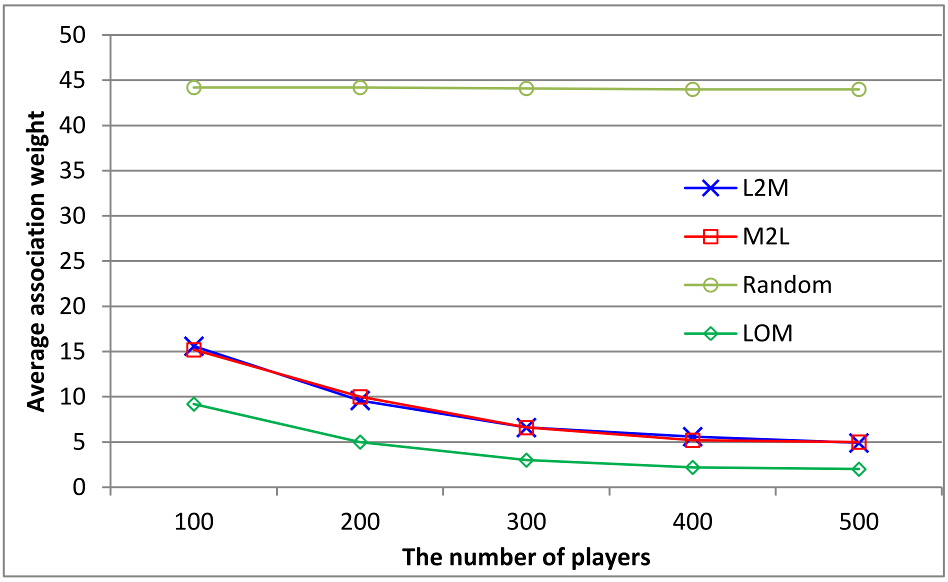

Figure 3. The average association of LOM and other matchmaking algorithms

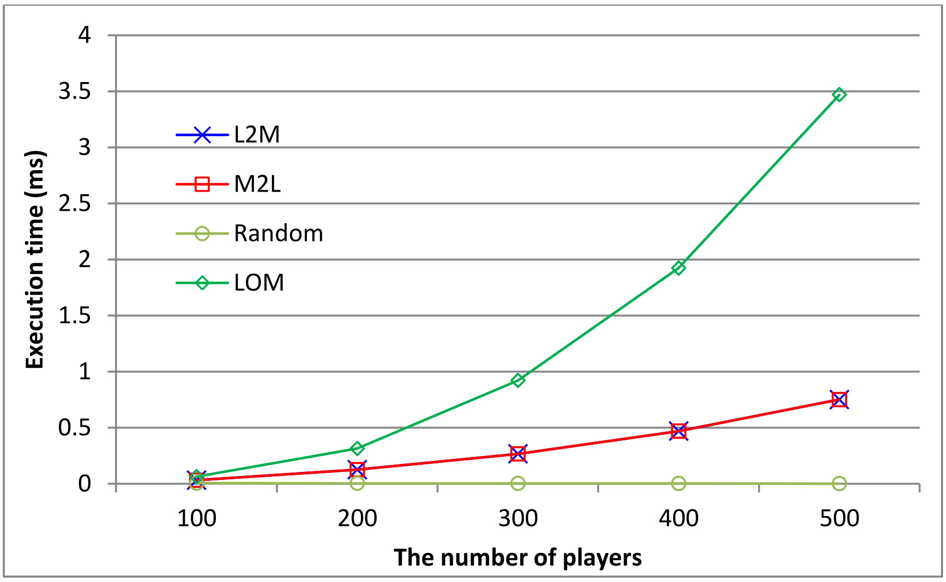

Figure 4. The execution time of LOM and other matchmaking algorithms

Fig. 4 is the comparison of execution time of LOM and other matchmaking algorithms. As expected, the random algorithm has the shortest execution time, since it just randomly select edges. L2M and M2L have almost the same execution time, since they are both greedy algorithms and have the same time complexity O(n^2), where n is the number of players for matchmaking. LOM spends more time than other algorithms, especially for games of larger scales. This is because the MCMF algorithm used by the LOM algorithm is based on the Edmonds-Karp algorithm [5], which is of time complexity O($V \cdot E^2$) for a graph whose node set is of size V and edge set is of size E. The time complexity of LOM is thus O(n^5). This is why LOM has the longest execution time. However, the time compleixty of LOM is still polynomial. Therefore, LOM is feasible for games of a moderate number of players.

5. CONCLUSION

Matchmaking is an important service to group players to start game sessions in modern MOGs. There are two types of matchmaking criteria: connection-based and skill-based. In the connection-based criterion, players with higher mutual network connection speed are grouped together. In the skill-based criterion, players with close skill ratings are grouped together. In this paper, we propose a new matchmaking criterion, association-based criterion, by which players with high association are grouped into a session of a game accommodating many independently running parallel sessions. We also design a fast and balanced matchmaking algorithm, called Leader Oriented Matchmaking (LOM), on the basis of the association-based criterion. By our evaluation, LOM outperforms other related algorithm in terms of the average association per session. LOM spends more time in computation than other algorithms. However, the time complexity of LOM is polynomial and thus LOM does not rebate players' gaming experiences if the scale of the game is not too large. In the future, we will focus on the problem about how to calculate the association degree between two MOG players more accurately and more efficiently.

6. REFERENCES

[1] S. Agarwal and J. R. Lorch, Matchmaking for online games and other latency-sensitive P2P systems, in *Proc. of ACM SIGCOMM Conference on Data Communication*, 315-326, 2009.

[2] C. H. Chien, S. Y. Hu, J. R. Jiang, and C. W. Cheng, Bandwidth-Aware Peer-to-Peer 3D Streaming, *International Journal of Advanced Media and Communication (IJAMC)*, 4(4), 324-342, 2010.

[3] R. Coulom, Whole-history rating: A Bayesian rating system for players of time-varying strength, in *Proc. of the 6th International Conference on Computers and Games (CG'08)*, 113–124, 2008.

[4] O. Delalleau, E. Contal, E. Thibodeau-Laufer, R. C. Ferrari, Y. Bengio, and F. Zhang, Beyond skill rating: advanced matchmaking in ghost recon online, *IEEE Trans. on Computational Intelligence and AI in Games*, 4(3), 167-177, 2012.

[5] J. Edmonds and R. M. Karp, Theoretical improvements in algorithmic efficiency for network flow problems, *Journal of the ACM*, 19(2), 248–264, 1972.

[6] T. Graepel and R. Herbrich, Ranking and matchmaking, *Game Developer Magazine*, 25–34, 2006.

[7] R. Herbrich, T. Minka, and T. Graepel, TrueSkill: a bayesian skill rating system, *Advances in Neural Information Processing Systems*, 569–576, 2006.

[8] S. Y. Hu, J. R. Jiang, and B. Y. Chen, Peer-to-Peer 3D Streaming, *IEEE Internet Computing*, 14(2), 54-61, 2010.

[9] J. R. Jiang, J. S. Chiou, and S. Y. Hu, Enhancing Neighborship Consistency for Peer-to-Peer Networked Virtual Environments, *Journal of Internet Technology*, 10(1), 29-36, 2009.

[10] J. Jimenez-Rodriguez, G. Jimenez-Diaz, and B. Diaz-Agudo, Matchmaking and case-based recommendations, in *Proc. of Workshop on Case-Based Reasoning for Computer Games*, 53–62, 2011.

[11] E. Laufer, R. C. Ferrari, L. Yao, O. Delalleau, and Y. Bengio, Stacked calibration of off-policy policy evaluation for video game matchmaking, in *Proc. of IEEE Conference on Computational Intelligence in Games (CIG)*, 1-8, 2013.

[12] P. L. Lanzi, D. Loiacono, and R. Stucchi, Evolving maps for match balancing in first person shooters, in *Proc. of IEEE Conference on Computational Intelligence and Games (CIG)*, 1-8, 2014.

[13] J. Manweiler, S. Agarwaly, M. Zhangy, R. R. Choudhury, and P. Bahlym, Switchboard: a matchmaking system for multiplayer mobile games, in *Proc. of the 9th International Conference on Mobile Systems, Applications, and Services (MobiSys'11)*, 71-84, 2011.

[14] J. Riegelsberger, S. Counts, S. Farnham, and B. C. Philips, Personality matters: incorporating detailed user attributes and preferences into the matchmaking process, in *Proc. of the 40th Annu. Hawaii Int. Conf. on System Sciences*, 2007.

[15] M. Véron, O. Marin, and S. Monnet, Matchmaking in multi-player on-line games: studying user traces to improve the user experience, in *Proc. of Network and Operating System Support on Digital Audio and Video Workshop*, 2014.

[16] R. C. Weng, and C. J. Lin, A Bayesian approximation method for online ranking, *Journal of Machine Learning Research*, 267–300, 2011.

[17] Call of Duty, website: https://www.callofduty.com/

[18] Dragon Age, website: http://www.dragonage.com/

[19] World of Warcraft, website: http://battle.net/wow/