Relay Strategy in Online Mobile Games: A Data-driven Approach

Guowei Zhu
zhuguowei@china-aii.com
China Academy of Industrial Internet

Kan Lv
lvkan@china-aii.com
China Academy of Industrial Internet

Ge Ma
mage@china-aii.com
China Academy of Industrial Internet

Weixi Gu*
guweixi@china-aii.com,guweixigavin@gmail.com
China Academy of Industrial Internet

ABSTRACT

With the booming of online mobile games (OMGs), game operators need to provide high-quality game service for users. Using relay has become the *de factor* approach for game streaming today, because it is easy to use (e.g., game sessions can be redirected via CDN servers) and has good scalability. Today, it has become the norm rather than the exception for game operators to hire CDN servers for their game services in a pay-per-use manner to serve massive users. Given the limited resource, selecting game sessions which are relayed has become a critical decision that can significantly affect users' quality of experience (QoE). Conventional strategies are generally rule-based, e.g., assigning game sessions to relay paths according to their past network performance, but cannot guarantee any particular QoE level because network performance dynamically changes. In this paper, we propose using data-driven approach to study network performance of game sessions in temporal and spatial patterns. Our findings indicate that there is obvious regularity for network performance of game sessions in temporal and spatial patterns. We design a machine learning-based predictive model to capture the quality of a game session given particular network performance metrics. Based on that, we strategically assign game sessions to relay paths to maximize the overall QoE. Trace-driven experiments are used to demonstrate the effectiveness and efficiency of our design.

CCS CONCEPTS

• **Networks → Network resources allocation**; *Network management*; *Network monitoring*.

KEYWORDS

Quality of Experience; Online Mobile Games; Network Measurement;Relay Network

ACM Reference Format:

Guowei Zhu, Kan Lv, Ge Ma, and Weixi Gu. 2020. Relay Strategy in Online Mobile Games: A Data-driven Approach. In *Adjunct Proceedings of the 2020 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2020 ACM International Symposium on Wearable Computers (UbiComp/ISWC '20 Adjunct), September 12–16,2020, Virtual Event, Mexico.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3410530.3414595

*Corresponding Author.

UbiComp/ISWC'20 Adjunct, 12–16 September 2020, Cancun,Mexico

ACM ISBN 978-1-4503-8076-8/20/09…$15.00
https://doi.org/10.1145/3410530.3414595

1 INTRODUCTION

With the integration of mobile internet and online games, online mobile games (OMGs) have become one of the most important multimedia entertainment services on the internet. It is reported that mobile games, as of 2019, have occupied 45% of the global gaming market[24] with the emergence of a large number of MOBA mobile games, including Honor of Kings[1], Contra[2], CrossFire[3], etc. The increasing popularity of OMGs brings about steep rise in online users. Compared with other multimedia services, e.g.,video and audio, OMGs require frequent interaction which makes them more sensitive to network conditions such as delay, packet loss, and jitter. To satisfy the massive users around the world, game operators deploy relay service, which provides users with better network performance to interact with game servers.

Although relay service proves to be an effective solution to improve network performance in OMGs streaming, it remains a critical problem how to efficiently allocate network resources under such paradigm, i.e., decisions on which game sessions should be relayed can significantly influence the level of Quality of Experience(QoE). An efficient assignment strategy can maximize the overall user experience in game service.

Conventional relay selection strategies[1, 27, 38], which are mostly rule-based, generally cannot dynamically select game sessions from the overall perspective. The reasons are as follows:

First, rule-based selection strategies fail to analyze users' network performance from the overall perspective. Under this scenario, game operators usually learn the rules from historical data, which would neglect the dynamic variability of network performance.

Second, a complex relationship between Quality of Service(QoS) and QoE has been demonstrated in previous studies[21, 22]. On one hand, there is interdependence among QoS metrics, e.g., loss rate and jitter rise when delay increases. On the other hand, context factors including static factors (e.g., AS, DeviceType, AP, Basestation) and temporal factors (e.g., slack hours, busy hours) can indirectly impact on QoS and QoE factors.

[1]http://pvp.qq.com
[2]https://hdl.qq.com
[3]http://cfm.qq.com

To address these problems, we propose a data-driven relay selection strategy. First, we propose a relay system architecture in which distributed CDN severs are taken as relay nodes. Then, we design a prediction model, which can accurately determine a game session quality with both QoS factors and context factors taken into account. Finally, we formulate the relay selection as an optimization problem maximizes the overall user experience in the relay service paradigm. Our contributions can be summarized as follows:

▹ We use a large-scale dataset from the most popular mobile game, Honor of Kings, covering over 120 million session traces which consist of more than 30 million users engaging in 25 million PvP combats. We analyze network performance of game sessions from temporal and spatial patterns. We observe that the occurrence of poor network presents chronical continuity and spatial extension at the AP/basestation level.

▹ With limited resources, we formulate the problem of relay selection as an optimization problem, and design a reinforcement learning-based algorithm to solve it. In our algorithm, we respectively calculate the benefits of relay or not in each game session. Game sessions with the highest quality improvement when relayed will be redirected to connect via CDN servers.

▹ We deploy trace-driven experiment to verify the effectiveness of our design. In particular, we configure the simulation environment based on parameters in our dataset. Compared with conventional approaches, our design can significantly improve the network performance of OMGs up to 30%.

The rest of the paper is organized as follows. Sec.2 surveys related work. Sec.3 presents measurement result in both temporal and spatial patterns. In Sec.4, we design game quality prediction model and propose relay strategy. We evaluate our performance of our design in Sec.5. Finally, we conclude the paper in Sec.6.

2 RELATED WORK

Applications of relay can potentially improve the quality of service (QoS) in multimedia services, and have been commonly used to provide various networking services[13, 15, 16, 36, 37]. The network performance could be improved significantly if a relay session is selected appropriately[20, 33]. Jiang et al.[20] propose to use predictive model to select a subset of telephone calls to be relayed using the relay network. Zhang et al.[36] design a two-phase approach to achieve efficient and accurate relay selection in peer-to-peer networks. Pang et al. [25] employ edge devices in the network as relays to improve the quality in crowdsourced live streaming. A problem in the mobile game streaming scenario is how to improve the game quality using relay paradigm. The limitation of conventional rule-based strategies is that they fail to consider complexity and dynamics of network, e.g., selecting relayed game sessions according to their past network performance metrics.

Quality of Experience (QoE) is a concept that involves subjectively perceived quality[19]. QoE indicates how users perceive the overall value of a service. Several research efforts have been devoted to understanding users' QoE in multimedia service, such as video and audio[7, 11, 23, 30, 34]. Traditional approach of measuring QoE is Mean Opinion Score(MOS)[34, 35]. Mok et al.[23] utilize MOS to express QoE and perform subjective experiments to evaluate how QoS correlates with QoE. Song et al.[31] design a service quality assessment framework named Q-score, that captured QoE in a timely fashion using offline learning and online computation. The above works generally used QoS factors and users' subjective feedback to infer users' QoE. Recently, the measurement of QoE have switched from the subjective method to objective ones[3, 18]. The metric of objective approach is based user's *engagement* in multimedia service[9, 10, 10, 14], e.g., watching time indicates the engagement of user in online video service[2], while call duration is regarded as user's satisfaction in VoIP service[8]. All of these objective metrics are based on the assumption that the higher the user experience, the longer the user will immerse in multimedia service. However, previous works[17, 21, 22] have demonstrated that user's behavior cannot be used to directly measure QoE in OMGs.

Appropriate QoE metric is the key to performing effective OMGs service. Previous effort research in game experience still use MOS approach[4–6, 12, 28, 29]. However, QoE in OMGs service is dynamically determined by game session quality. In our study, we propose to use *game session quality* as user experience in game streaming.

3 DATASET AND MEASUREMENT

In this section, we first give a brief introduction of OMGs. Then, we present our data collection and dataset. Finally, we describe measurement studies conducted to discover network performance in temporal and spatial patterns.

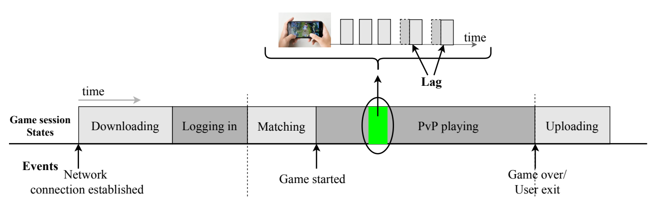


Figure 1: An illustration of a game session life and associated with client-side events.

3.1 Background and Data Collection

3.1.1 Background. As shown in Fig. 1, in a game session, the client-side experience downloading updated files, logging in, matching with other players, PvP (Player versus Player) playing, and uploading records. The active network probing continues in matching and PvP process. Meanwhile, using the SDK embedded in the game, we can collect rich information on client-side and report the records when the game session is over. Actually, the game users interact with is a series of frames rendered on screen. The client-side synchronize the user operations to the game server via Internet in real time, and then render the screen based on the instructions returned by game server as well as frame rate settings on client-side. During PvP process, game server periodically instructs to all sessions belonging to the same PvP so that all matched players can play game together. If the network or mobile phone's performance goes down, there will be a temporary "freeze" in rendering, which is called *lag* in games. Regardless of game genres and user skills, the user will have a bad experience when he encounters the game lag.

3.1.2 Data Collection. Our dataset is collected from *Honour of Kings*, the most popular mobile game created by Tencent. There are more than 500 million session traces in one day, limited by computing power, we randomly sample 120 million session traces, which consist of over 30 million users engaging 25 million PvP combats on September 11, 2017. The details of session traces from Honour of Kings are follows.

Network performance metrics: The client-side periodically sends probing packets to detect the round-trip time (RTT) experienced by users. Specifically, the probing periods in matching and PvP processes are 3s and 5s, respectively. Utilizing these RTTs, we can measure network performance from the following nine aspects in a session: delay, jitter, loss rate, jump ratio, R_0^{100} (Ratio of delay between 0ms and 100ms), R_{100}^{200} (Ratio of delay between 100ms and 200ms), R_{200}^{300} (Ratio of delay between 200ms and 300ms), R_{300}^{460} (Ratio of delay between 300ms and 460ms), R_{460}^{∞} (Ratio of delay greater than 460ms).

Network access information: In each trace, the client-side is associated with the following network access information: Internet Service Provider (ISP), Autonomous System (AS), Access Point (AP) and basestation.

Client-side information: Each trace contains the following client-side information: (1) Game session quality, we calculate the fraction of delayed frames in all rendering frames as *lag ratio*, which is used to capture the quality in a game session. (2) Device information, including device type (e.g., Apple, Huawei etc). During PvP process, we further detect the battery level, CPU and memory usage in 5s cycles. (3) Temporal information, including time of day (e.g., slack hours, busy hours).

In our dataset, each trace records the above information. Among them, we use game session quality as user's experience metric in OMGs service. In term of context factors, network access information, device information and temporal information are considered, which will be discussed in Sec.5.1.2.

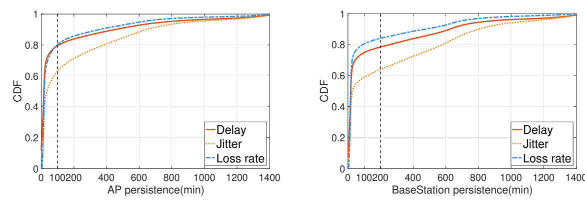
3.2 Measurement

Based on the collected traces, we can utilize the dataset to mine the network performance in temporal and spatial patterns. To this end, we first present the access mode in OMGs service.

3.2.1 Access Mode Analysis. As last-mile in mobile game streaming, the network performance between AP/basestation and client is a key factor impacting user experience[26, 32]. Understanding network performance from the view of AP/basestation is the first step towards designing relay strategies. In our dataset, 60% (40%) of game sessions are connected via AP (basestation), including over 22 million APs and 10 million basestations. We measure the associated game session counts of each AP/basestation, and observe that there are more than 80% (70%) of APs (basestations), which are associated with less than 5 game sessions. This phenomenon indicates that the access mode of game sessions are uniformly distributed.

To study the impact of network performance in online game streaming, we define poor network rate of game sessions. First, when the network performance of a game session meets any of the following conditions: *delay* $\geq 200ms$, *jitter* $\geq 100ms$ and *loss rate* ≥ 0.1, we categorize it as *poor network session.* And then, we calculate the fraction of poor network sessions in a set of game sessions as *Poor Network Rate (PNR).*

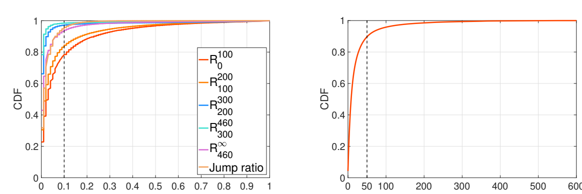
3.2.2 Network Performance in Temporal Patterns. In this section, we describe measurement studies conducted to discover network performance in temporal patterns. To ensure the rationality of statistic, we choose the APs/basestations, which are associated with over 5 game sessions in our dataset. Fig. 2 (a) and (b) respectively plot the cumulative distribution functions (CDF) of poor network duration in scenario of AP and basestation. We observe that poor network duration of AP and basestation is long-tail distribution. For example, less than 30% (40%) of APs (basestations) continue to suffer poor network for above 100mins (200mins), while more than 60% of APs and basestations sustain poor network for below 10mins. These results indicates that conventional rule-based relay strategies fail to capture the dynamics of network, we need to select relayed sessions based on active probing in real time.



(a) AP (b) Basestation

Figure 2: CDF of poor network duration.

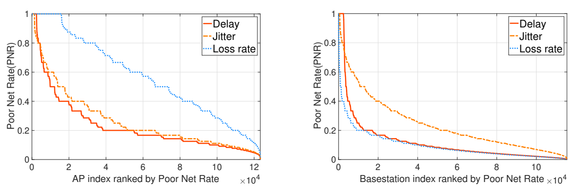
In matching and PvP processes, the client-side periodically sends probing packets to detect network performance, including direct and relay path. Since 80% of game sessions have durations between $10 \sim 30$ minutes[22], we measure the difference of network performance metrics between matching and PvP process, as shown in Fig. 3. We observe that network performance metrics are slightly different in matching and PvP processes, e.g., there are 90% of game sessions whose absolute value of difference in delay distribution (delay) is less than 0.1 (50ms). The above observations indicate that we could respectively estimate the game qualities via direct and relay path using network performance metrics in matching process, based on game quality prediction model which will be described in Sec.4.3.



(a) Network Performance (b) Delay

Figure 3: CDF of network performance difference between matching and PvP process.

3.2.3 Network Performance in Spatial Patterns. We now analyze network performance in spatial patterns. We conduct this analysis by calculating the PNR of each AP/basestation, which is associated with over 5 game sessions in our dataset. Fig. 4 (a) and (b) respectively show the PNR of APs and basestations ranked in descending order. We observe that the PNR of APs is uniform distributed, while the PNR of basestations is long-tail distributed, e.g., 90% of basestations have PNR of less than 20%. This phenomenon indicates that network performance differs in spatial patterns, and we need to perform relay selection from global perspective.



(a) Network Performance (b) Delay

Figure 4: PNR of APs and basestations ranked in descending order.

3.2.4 Analysis Result. The key observations from this section are:

- *Wide-area* distribution for game sessions, e.g., there are over 30 million APs and basestations associated with game sessions in one day.
- Game sessions suffering from poor networks performance are spread spatially and temporally.

These observations motivate the need for a relay network that provides better paths with a *global* view of game sessions, and the need to choose relayed sessions *selectively and dynamically*.

4 GAME QUALITY-AWARE RELAY STRATEGY

In this section, we design a relay strategy for OMGs. We use geographically distributed CDN servers to deploy relay service and a centralized control method to select which games sessions to relay.

We first present a relay architecture and define the problem. Then, we design a data-driven game quality prediction model. Finally, we propose a relay selection strategy which is based on reinforcement learning.

4.1 Relay System Architecture

Fig. 5 plots a schematic illustration of the relay system architecture for OMGs. The relay service uses CDN servers distributed in different geographic locations, such as Alibaba Cloud, Tencent Cloud, China Cache, etc., as relay nodes. Connected through a high-bandwidth network, these relay nodes show the characteristics of broader bandwidth and lower delay, which enable them to significantly improve network performance.

All data in a game session are transmitted through the established path, which is either "direct path" shown by blue arrow or "relay path" shown by red arrow. The connection of each game session will first "pass" the intercity route built by ISP operators. Then the paths between clients and relay nodes are determined by BGP.

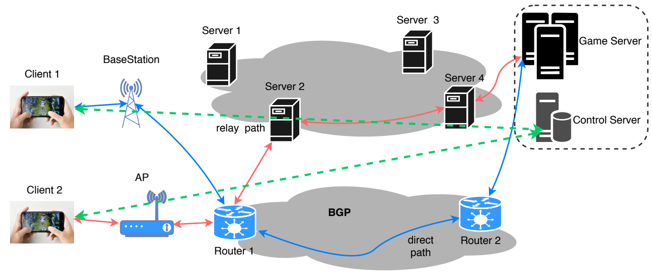


Figure 5: Relay architecture with relay nodes at globally distributed servers. A game session can either take "direct path" (blue arrow) or a "relay path" (red arrow).

In each game session, client-side periodically sends probing packets to the game server through "direct path" or "relay path" in the matching process, and collects network performance metrics including delay, then reports to the control server. Based on the information uploaded from client, model and restrictions (e.g., bandwidth, server load, etc.), the control server will determine whether the session is connected through "direct path" or "relay path" and which relay nodes are involved in the connection. In addition, the control server needs to conduct real-time monitoring on all relay nodes in order to keep informed their state (e.g., bandwidth, load, etc.). To avoid over-limit load of control server, we can apply for multiple control instances. The specific method is discussed in Sec. 4.4.

4.2 Problem Statement

The problem scenario is to assign a "direct path" or "relay path" to each game session. We use the symbol S to represent the set of game sessions that need to be assigned, and the symbol C to represent the set of options ("direct path" and "relay path"). Correspondingly, $s \in S$ and $c \in C$ represent a game session and a path respectively. $Q(s, c)$ is represented for the expected game quality when option c is selected for session s, the game quality model will be discussed in Sec. 4.3. Since we use the *lag ratio* of a game session as game quality metric, the smaller value of $Q(s, c)$ means the better user experience. In our study, we assume that the path options of each session in set S are mutually independent, i.e., the performance of each session will not be affected whether other sessions are relayed or not.

The purpose of designing relay strategy is to assign connection paths for $s \in S$. We use $F : S \to C$ to indicate the allocation rules output by the algorithm and $F(S)$ to indicate the connection path assigned to $s \in S$. Our optimization goal is to find an optimal allocation rule, as shown in the following formula:

$$arg \min_{F \in C^S} \sum_{s \in S} Q(s, F(s)) \tag{1}$$

Therefore, this problem is actually an optimization with the minimum objective, which is using algorithm to find the optimal solution in the solution space.

4.3 Game Quality Prediction Model

From a high-level overview, game quality can be formulated as the function of network performance metrics, i.e., *Lag ratio* $=$

$f(Metric_i)$. *Lag ratio* denotes the measurement of game quality and $Metric_i$ denotes the network performance metric, e.g., delay, loss rate, jump ratio, etc. There are two core requirements for designing a model which can predict game quality. 1) The function relationship f needs to meet the requirements of accuracy, efficiency and interpretability, so that game operators can accurately measure the game quality in real time. 2)The model should enable dynamic learning and provide guidance for relay mechanism. However, multiple challenges exist in meeting these requirements:

- *Complex relationships:* Previous research work[22] found that the relationship between network performance and game quality was rather complicated. For example, there is non-monotonicity between delay and lag ratio, for longer delay will cause serious packet loss in data transmission which will reduce the number of new rendered frames. Simultaneously, when jitter is below 250ms, there is an obvious positive linear correlation between jitter and lag ratio. When jitter exceeds 250ms, the lag ratio drops rapidly.
- *Interaction between network performance metrics:* Network performance metrics are interdependent and influential to each other, e.g., longer delay will lead to the increase of packet loss rate.
- *Impact of context factors:* Implicit context factors also have impacts on network performance and game quality, directly or indirectly, such as temporal factors (e.g., slack or busy), access factors (e.g., AS, AP, Basestation, etc.), device factors (e.g., device type, operating system, etc.).

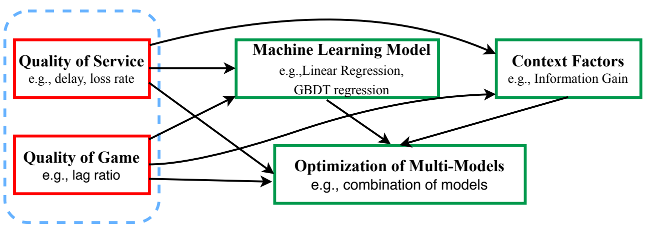


Figure 6: High-level road map of design game quality prediction model.

To capture above challenges, we design a game quality prediction model. Fig. 6 shows the road map of our approach. We first build a basic prediction model using a typical machine learning algorithm, then extract context factors that affect network performance metrics and game quality. After determining the implicit context factors, we optimize the game quality prediction model. The specific solutions are as follows:

- *Solve complex relationships and interdependencies among factors:* We employ machine learning algorithm as a functional mapping between network performance metrics and game quality. In choosing the machine learning algorithm, we follow three requirements. 1) It can solve the interdependence problem between various network performance metrics, which algorithms that hold independent assumption fail to solve. 2) It can deal with the nonlinear relationship between network performance metrics and game quality, which the conventional linear regression algorithms fail to solve the problem. 3) It has the characteristics of accuracy, efficiency and interpretability, which overwhelmingly complex and unpractical algorithms, such as deep learning algorithms, fail to present. We evaluate candidate models in Sec. 5.1.1.
- *Identify important context factors:* Dynamic context factors, such as CPU usage rate, memory usage rate and power consumption, have impacts on network performance metrics and game quality. We first classify context factors from the perspective of time attributes, connection attributes and device attributes, then use the relative information entropy to analyze their impact. This part will be discussed in Sec. 5.1.2.

4.4 Relay Strategy

In this paper, the relay strategy we proposed is based on the assumption that there's slightly difference of network performance metrics in matching and PvP process. The measurements in Sec. 3.2.2 also testify the validity of this assumption. Therefore, we can use metrics collected during matching process to *estimate* the game quality. Since network performance metrics in both direct and relay path are actively probed in parallel by client-side during the matching process of each game session, game quality under the both paths can be predicted respectively.

As to achieve our goal of finding an optimal assignment rule, there are two optional solutions:

1) *Data-based prediction:* After collecting data that uploaded from client-side and conducting analysis and modeling, we use machine-learning algorithm to predict game qualities in different connection paths. Then we choose the path that leads to the best game quality.
2) *Search solution space:* We calculate the game quality of each session in the set S, i.e., collection of sessions to be assigned, under each scenario of path selection. Then we search the assignment rule F that validates formula (1);

However, it's difficult to derive the optimal allocation rule from the above solutions. First, using data-based prediction approach, we need to collect data, train model and predict game quality every tens of minutes. According to our measurement and analysis result in Sec.3.2, network performance shows dynamic variability in temporal and spatial patterns. The model trained tens of minutes ago may not adaptable to the current network condition. Second, using search-based solution space approach, we need to traverse all the options of each unassigned game session. Although the optimal allocation rule may be found, the process results in a huge search space with $|S|$ game sessions and $|R|$ options producing $\Theta(|R|^{|S|})$allocation rules, which seriously affects the efficiency of decision-making. Thus, data-based prediction and search-based detection both have shortcomings with the former one leads to inaccurate assignment and the latter one will lower the efficiency of decision-making.

In this paper, we proposed an approach that combines prediction and exploration-exploitation. Fig. 7 illustrates the process of relay selection strategy and Algorithm 1 demonstrates its pseudo code. The detailed logic is as follows:

1. Obtain the latest trained game quality prediction model *Pred*, exploration probability ϵ, threshold η;
2. Initialize assigned strategy F as null;

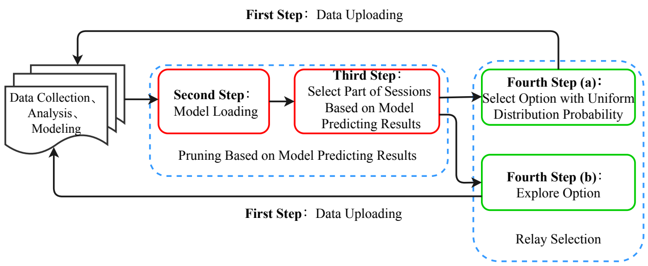


Figure 7: Overview of relay selection that combines prediction and exploration-exploitation.

Algorithm 1 Relay Selection Algorithm

Require: Set of game sessions S to be assigned to relaying options C, exploration probability ϵ and threshold η
Ensure: Assignment F

procedure RelaySelection(S, C, ϵ, η)
  $F \leftarrow \emptyset$
  for $s \in S$ **do**
    $Diff \leftarrow \mid Pred_{direct}(s) - Pred_{relay}(s) \mid$
    if $Diff \geq \eta$ **then**
      $F(s) \leftarrow arg \min_{f \in \{direct, relay\}} Pred_f(s)$
    else
      if $RandomFloat(0, 1) \leq \epsilon$ **then**
        $c \leftarrow$ Explore($s, C, F, Pred$)
        $F(s) \leftarrow c$
      else
        $F(s) \leftarrow Random(C)$
      end if
    end if
  end for
  return F
end procedure

3. For each session in set S, we respectively calculate game quality under each option (direct or relay) and then calculate the difference $Diff$ between the two options;
4. When the value of $Diff$ exceeds the pruning threshold η, we assign the game session s with the path, which makes the smallest value of $Pred$;
5. When the value of $Diff$ is under the pruning threshold η, we employ ϵ-Greedy Algorithm, using the probability of ϵ to conduct exploration, the pseudo code of which is demonstrated in Algorithm 2, and the probability of $1 - \epsilon$ to conduct random assignment.

In the whole process, the game quality model is built in a certain time cycle, i.e. usually $30 - 60$mins, while data uploading and relay selection are carried out in real time.

Prediction-based pruning: We narrow the search space of exploration algorithm according to prediction results to improve decision efficiency. We first define the threshold η. If the absolute value of the difference between game qualities in the two options exceeds the threshold η, we choose the connection path that leads to lower prediction result. If the absolute value is less than the threshold η, we conduct reinforcement learning-based algorithm to select the optimal path.

Algorithm 2 Exploration Algorithm of Relay Selection using UCB1

Require: A game session s to be assigned, relaying options C, a relay assignment F and game quality model $Pred$
Ensure: Assignment of game session s

function Explore($s, C, F, Pred$)
  $\text{ucb}_{\text{min}} \leftarrow \infty; c_{best} \leftarrow null$
  $w \leftarrow \frac{1}{|C|} \sum_{c \in C} (Pred_c(s))$
  $T \leftarrow \mid C \mid +1$
  for $c \in C$ **do**
    $S_c \leftarrow \{s' \mid F(s') = c\}$
    $\text{ucb} \leftarrow \frac{1}{w|S_c|} \sum_{s' \in S_c} Pred_c(s) - \sqrt{\frac{0.1 \log T}{|S_c|}}$
    if $\text{ucb} < \text{ucb}_{\text{min}}$ **then**
      $c_{best} \leftarrow c$
      $\text{ucb}_{\text{min}} \leftarrow \text{ucb}$
    end if
  end for
  return c_{best}
end function

Exploration-exploitation-based selection algorithm: We formula the selection of session paths as a "maximum single-step reward" problem with set C as the collection of "actions" and game quality under every selection as "reward". Given the dynamic changes of game quality related to each session, we employ exploration-exploitation approach in this paper. Exploration is carried out using Upper Confidence Bound Algorithm (UCB1), with the probability of ϵ, to select the paths of partial game sessions. As to the paths of the rest game sessions, we conduct exploitation via random algorithm with the probability of $1-\epsilon$. Algorithm 2 illustrates the pseudo code of exploration-exploitation algorithm.

Resource restrictions: In a real system environment, only a certain percentage of game sessions can be relayed, e.g., 10%, since the limited sever and bandwidth resources. Under these constraints, we can choose to relay the game sessions that best improve game quality, e.g., top 10% game sessions.

5 EVALUATION

In this section, we conduct experiments using traces from our dataset to verify our design and evaluate its performance.

We first compare accuracies of several typical machine learning algorithms and choose the optimal one. Then, we optimize game quality prediction model after identifying important context factors. Finally, we evaluate the efficiency of our design.

5.1 Quality Model Prediction

5.1.1 Model Accuracy. We cast the problem of modeling the relationship between network performance metrics and game quality as a continuous regression problem. In this game quality prediction, we first compare six classical machine learning algorithms that are widely used in real-world systems, to evaluate the performance of the different algorithms. The machine learning algorithms tested are the *Linear Regression (LR), Classification And Regression Tree (CART), Support Vector Machine (SVM), Random Forest, Adaboost and Gradient Boosted Decision Tree (GBDT)*, and we evaluate these algorithms from the following performance indicators, *Mean Absolute Error(MAE), Mean Square Error(MSE), Root Mean Square Error(RMSE), Correlation Coefficient(Correlation).* Based on the dataset used in the measurement part, we perform 10-fold cross validation and summarize the performance of the different algorithms in table 1. From table 1, we can see that the Gradient Boosted Decision Tree (GBDT) algorithm achieves better performance. Meanwhile, GBDT is sufficiently expressive to capture the complex relationships. We further use the GBDT to predict the game quality in a session.

Table 1: Prediction results.

Regression Algorithm	MAE	MSE	RMSE	Correlation
LR	0.051	0.015	0.123	0.478
CART	0.063	0.029	0.171	0.308
SVM	0.118	0.025	0.159	0.116
Random Forest	0.051	0.013	0.117	0.555
Adaboost	0.107	0.020	0.141	0.493
GBDT	**0.045**	**0.012**	**0.110**	**0.612**

5.1.2 Context Factors Analysis. We identify the following three categories of context factors from our dataset:

- *Temporal factors:* This refers to *time of day* of game sessions (e.g., $00:00{\sim}09:59$, $10:00{\sim}17:59$, $18:00{\sim}23:59$).
- *Access factors:* This includes the following factors associated with a game session, *Autonomous System* (AS, e.g., Beijing Telecom, Guangdong Unicom etc), *wireless type* (wifi or cellular), *AP* and *basestation.*
- *Device factors:* This refers to *mobile phone model* (e.g., iphone, huawei etc).

We use *information gain analysis* to identify the hidden relationships between context factors and network performance metrics and game quality. Table 2 presents the information gain between each factor with respect to the nine network performance metrics and game quality. In our study, we make the context factors whose relative information gains are more than 5% as significant factors.

Our observations are as follows: (1) The main context factors are AP, basestation and AS. (2) Our results also reconfirm prior observations that APs and basestations significantly impact network performance and game quality in OMGs.

5.1.3 Optimization of Multi-Models. To incorporate the context factors into the prediction model, we propose the following two candidate approaches:

- *Add context factors as new features:* The simplest approach is to add the key context factors as additional features to the machine learning algorithm and retrain the prediction model.

Table 2: Relative information gain(%) between different context factors and network performance metrics and game quality.

Quality Metrics	AS	Wireless type	Time of day	Device type	AP	Basestation
Delay	5.23	2.99	0.04	3.75	96.99	83.67
Jitter	1.46	2.32	0.07	3.78	97.08	83.34
Jump ratio	1.31	1.06	0.16	2.61	96.79	79.66
Loss rate	1.72	2.82	0.23	3.51	96.70	81.32
R_0^{100}	3.32	2.07	0.05	2.96	96.98	82.97
R_{100}^{200}	3.54	3.37	0.03	2.63	96.99	82.90
R_{200}^{300}	2.96	1.46	0.14	3.51	96.84	79.21
R_{300}^{460}	4.23	1.48	0.17	4.98	96.78	79.52
R_{460}^{∞}	2.31	1.23	0.17	3.52	96.76	80.65
Lag ratio	7.73	1.94	5.20	22.23	47.34	45.75

- *Split dataset based on context factors:* Another approach is to split the dataset based on the context factors (e.g., AS) and separately train models for each split. Our prediction model would then be the union of multiple single models, i.e., one for each combination of the values of various context factors.

Both approaches have merits and demerits. The feature-addition approach has the characteristic of simplification and requiring slight modifications to the machine learning framework. Specifically, it will train a single unified model over the dataset. The augmented model might be less intuitive and makes it harder for game operators to visually reason about the network performance and game quality for system design. In contrast, the split dataset approach will generate intuitive expression by combining the context factors. The challenge of the split approach is the "data sparseness", i.e., as we have more context factors to split, the available data in each split becomes progressively sparser. Therefore, the model trained may not have sufficient data samples to build a robust model. Fortunately, we have two reasons to be hopeful in our system design. First, we only split the dataset based on a few key context factors. Second, with the booming of OMGs, we will have larger datasets to train models and that will alleviate concerns with limited data in each split.

We compare the improvements of model performance using the two approaches. For this study, we use AS as the key context factor and compare the increase in performance using the feature-addition and the split dataset approach. As shown in table 3, splitting based on the AS gives better improvement compared to adding AS as a feature. In our study, we employ split dataset approach to optimize our prediction model.

Table 3: Comparing feature-addition and split approach for the AS context factor.

Approach	MAE	MSE	RMSE	Correlation
Single model	0.045	0.012	0.110	0.612
Feature-addition	0.041	0.010	0.105	0.658
Split	**0.037**	**0.008**	**0.098**	**0.697**

5.2 Performance Evaluation

5.2.1 Experiment Setup. We develop an event-driven simulation platform that take users' playing mobile game activities and relay

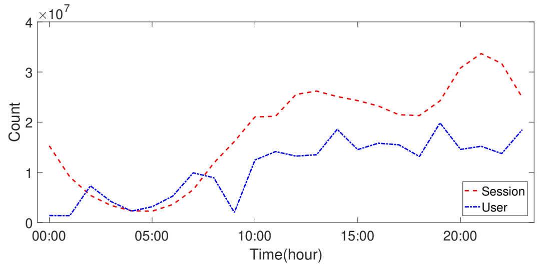


Figure 8: The changes of online game sessions and users in one day.

decisions as events to drive the experiments. We set up our experiments using the collected traces and compare our design with the other two baseline approaches.

Simulation environment: In the simulation experiment, we first generate 3307 ASes and over 10 million basestations and 22 million APs according to the dataset described in Sec.3.2, among which ASes are randomly distributed. In our study, we assume that BGP connections between ASes and game servers are relatively stable. Thus, delays between ASes and game servers are uniformly and randomly distributed by the interval of 0 to 150ms. The delays between APs/basestations and ASes are assigned base on the delay distribution in our dataset, and the "noise disturbance" is added over time.

Relay sever: 40 relay servers are generated and randomly associated with the 3307 ASes. We assume that the delays between AS routers and game servers via relay servers are uniformly and randomly distributed by the interval of 0 to 100ms, due to the high-bandwidth connection among relay nodes.

Game session request: Fig. 8 shows the changes of online game sessions and users in one day. Based on those changes, we conduct simulation on game session request in one day (24 hours). For each game session, we assign it either a basestation with the probability of 0.6 or a AP with the probability of 0.4. The delay of a game session should be the sum of the delays from client to AP/basestation, from AP/basestation to AS router, and from AS router to game server including "direct" and "relay" path. Among them, the delay between AP/basestation and client is randomly generated from 0 to 50ms. Based on the dataset in our study, we can also build the relationship between delay and network performance metrics including loss rate, jitter and jump ratio, etc.

Network control: To serve the simulated request queue of game sessions, the network control system updates game quality model every 30 minutes. The relay selection schedule game sessions in real time.

Baseline Approach: We compare our design with two other approaches, which are a prediction-based approach and a random selection approach. In the prediction-based approach, logistic regression model is used to make relay prediction.

5.2.2 Design Efficiency. *Poor network rate:* Fig. 9 demonstrates the poor network rate of game sessions under three different approaches, i.e., the relay strategy proposed in this paper, prediction-based strategy and randomly selection strategy. Based on all game sessions in the simulation experiment, we calculate poor network rates of three network performance metrics, including delay, loss rate and

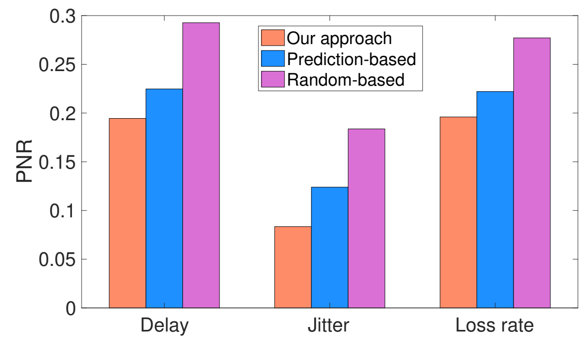
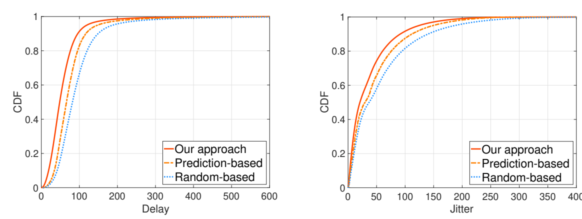


Figure 9: The PNR of game sessions under three different approaches.

jitter etc, by the three selection strategies mentioned above. As is shown in Fig. 9, our relay selection strategy reduces poor network rate by 30% ~ 40% compared with the other two strategies.



(a) Delay (b) Jitter

Figure 10: CDF of delay and jitter under three different approaches.

Improvement of network performance: We evaluate the network performance under the three strategies. Fig. 10 shows the distribution of delay and jitter under three different selection approaches. We can observe the relay selection strategy proposed in our paper averagely reduces delay and jitter by 30%, which contributes to a significant improvement of user experience.

6 CONCLUSION

It has become the norm for game operators to use a relay network paradigm in their online game service. Selecting game sessions which are relayed via CDN servers is key to provisioning a satisfactory QoE. In this paper, we propose a novel data-driven approach to select relayed game sessions in online game streaming. By conducting extensive measurements of traces from one of the largest online mobile games in the world, we identify that game sessions suffering from poor network performance are spread spatially and temporally. Based on these measurement insights, we design a learning-based game quality model to predict quality in game sessions. We then formulate our objective as an optimization problem and design a reinforce-learning-based algorithm to solve it. Extensive experiments driven by real-world traces demonstrate the effectiveness and and superiority of our design.

REFERENCES

[1] Mehdi Ghamari Adian and Hassan Aghaeinia. 2016. A two-level cooperative game-based approach for joint relay selection and distributed resource allocation in MIMO-OFDM-based cooperative cognitive radio networks. *Transactions on Emerging Telecommunications Technologies* 27, 1 (2016), 34–50.

[2] Athula Balachandran, Vyas Sekar, Aditya Akella, Srinivasan Seshan, Ion Stoica, and Hui Zhang. 2013. Developing a predictive model of quality of experience for internet video. In *ACM SIGCOMM Computer Communication Review*, Vol. 43. ACM, 339–350.

[3] Jordi Mongay Batalla, Piotr Krawiec, Andrzej Beben, Piotr Wisniewski, and Andrzej Chydzinski. 2016. Adaptive video streaming: Rate and buffer on the track of minimum rebuffering. *IEEE Journal on Selected Areas in Communications* 34, 8 (2016), 2154–2167.

[4] Justus Beyer, Richard Varbelow, Jan-Niklas Antons, and Sebastian Möller. 2015. Using electroencephalography and subjective self-assessment to measure the influence of quality variations in cloud gaming. In *Quality of Multimedia Experience (QoMEX), 2015 Seventh International Workshop on*. IEEE, 1–6.

[5] Wei Cai, Ryan Shea, Chun-Ying Huang, Kuan-Ta Chen, Jiangchuan Liu, Victor CM Leung, and Cheng-Hsin Hsu. 2016. A survey on cloud gaming: Future of computer games. *IEEE Access* 4 (2016), 7605–7620.

[6] Kuan-Ta Chen, Yu-Chun Chang, Hwai-Jung Hsu, De-Yu Chen, Chun-Ying Huang, and Cheng-Hsin Hsu. 2013. On the quality of service of cloud gaming systems. *IEEE Transactions on Multimedia* 16, 2 (2013), 480–495.

[7] Kuan-Ta Chen, Chun-Ying Huang, Polly Huang, and Chin-Laung Lei. 2006. Quantifying Skype user satisfaction. In *ACM SIGCOMM Computer Communication Review*, Vol. 36. ACM, 399–410.

[8] Kuan-Ta Chen, Chun-Ying Huang, Polly Huang, and Chin-Laung Lei. 2006. Quantifying Skype User Satisfaction. In *Proceedings of the 2006 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications (SIGCOMM '06)*. ACM, 399–410.

[9] Florin Dobrian, Vyas Sekar, Asad Awan, Ion Stoica, Dilip Joseph, Aditya Ganjam, Jibin Zhan, and Hui Zhang. 2011. Understanding the impact of video quality on user engagement. In *ACM SIGCOMM Computer Communication Review*, Vol. 41. ACM, 362–373.

[10] Weixi Gu, Ming Jin, Zimu Zhou, Costas J Spanos, and Lin Zhang. 2016. Metroeye: smart tracking your metro trips underground. In *Proceedings of the 13th International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services*. 84–93.

[11] Weixi Gu, Zheng Yang, Longfei Shangguan, Wei Sun, Kun Jin, and Yunhao Liu. 2014. Intelligent sleep stage mining service with smartphones. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*. 649–660.

[12] Weixi Gu, Yuxun Zhou, Zimu Zhou, Xi Liu, Han Zou, Pei Zhang, Costas J Spanos, and Lin Zhang. 2017. Sugarmate: Non-intrusive blood glucose monitoring with smartphones. *Proceedings of the ACM on interactive, mobile, wearable and ubiquitous technologies* 1, 3 (2017), 1–27.

[13] Osama Haq, Cody Doucette, John W Byers, and Fahad R Dogar. 2019. Judicious QoS using cloud overlays. *arXiv preprint arXiv:1906.02562* (2019).

[14] Miao He, Weixi Gu, and Ying Kong. 2017. Group recommendation: by mining users' check-in behaviors. In *Proceedings of the 2017 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2017 ACM International Symposium on Wearable Computers*. 65–68.

[15] Miao He, Weixi Gu, Ying Kong, and Lin Zhang. 2019. Identifying causal patterns from mobile sensing data: a case study on blood glucose inference. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. 77–80.

[16] Miao He, Weixi Gu, Ying Kong, Lin Zhang, Costas J Spanos, and Khalid M Mosalam. 2019. CausalBG: Causal Recurrent Neural Network for the Blood Glucose Inference with IoT Platform. *IEEE Internet of Things Journal* 7, 1 (2019), 598–610.

[17] Miao He, Weixi Gu, Yuxun Zhou, Ying Kong, and Lin Zhang. 2019. Causal feature selection for physical sensing data: a case study on power events prediction. In *Adjunct Proceedings of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*. 565–570.

[18] Te-Yuan Huang, Ramesh Johari, Nick McKeown, Matthew Trunnell, and Mark Watson. 2014. A buffer-based approach to rate adaptation: Evidence from a large video streaming service. In *Proceedings of the 2014 ACM conference on SIGCOMM*. 187–198.

[19] International Organization for Standardization 2009. *Ergonomics of human system interaction - Part 210: Human-centered design for interactive systems (formerly known as 13407)*. International Organization for Standardization.

[20] Junchen Jiang, Rajdeep Das, Ganesh Ananthanarayanan, Philip A Chou, Venkata Padmanabhan, Vyas Sekar, Esbjorn Dominique, Marcin Goliszewski, Dalibor Kukoleca, Renat Vafin, et al. 2016. Via: Improving internet telephony call quality using predictive relay selection. In *Proceedings of the 2016 ACM SIGCOMM Conference*. ACM, 286–299.

[21] Yuan Meng, Shenglin Zhang, Zijie Ye, Benliang Wang, Zhi Wang, Yongqian Sun, Qitong Liu, Shuai Yang, and Dan Pei. 2019. Causal Analysis of the Unsatisfying Experience in Realtime Mobile Multiplayer Games in the Wild. In *2019 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 1870–1875.

[22] Chou Mo, Guowei Zhu, Zhi Wang, and Wenwu Zhu. 2018. Understanding gaming experience in mobile multiplayer online battle arena games. In *Proceedings of the 28th ACM SIGMM Workshop on Network and Operating Systems Support for Digital Audio and Video*. 25–30.

[23] Ricky KP Mok, Edmond WW Chan, and Rocky KC Chang. 2011. Measuring the quality of experience of HTTP video streaming. In *12th IFIP/IEEE International Symposium on Integrated Network Management (IM 2011) and Workshops*. IEEE, 485–492.

[24] NewZoo. 2019. Global Games Market Report. NewZoo.

[25] Haitian Pang, Zhi Wang, Chen Yan, Qinghua Ding, Kun Yi, Jiangchuan Liu, and Lifeng Sun. 2018. Content Harvest Network: Optimizing First Mile for Crowdsourced Live Streaming. *IEEE Transactions on Circuits and Systems for Video Technology* 29, 7 (2018), 2112–2125.

[26] Changhua Pei, Youjian Zhao, Guo Chen, Ruming Tang, Yuan Meng, Minghua Ma, Ken Ling, and Dan Pei. 2016. WiFi can be the weakest link of round trip network latency in the wild. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE, 1–9.

[27] Xingyu Qian and Hao Wu. 2012. Mobile relay assisted handover for LTE system in high-speed railway. In *2012 International Conference on Control Engineering and Communication Technology*. IEEE, 632–635.

[28] Peter Quax, Anastasiia Beznosyk, Wouter Vanmontfort, Robin Marx, and Wim Lamotte. 2013. An evaluation of the impact of game genre on user experience in cloud gaming. In *Games Innovation Conference (IGIC), 2013 IEEE International*. IEEE, 216–221.

[29] Steven Schmidt, Saman Zadtootaghaj, and Sebastian Möller. 2017. Towards the delay sensitivity of games: There is more than genres. In *Quality of Multimedia Experience (QoMEX), 2017 Ninth International Conference on*. IEEE, 1–6.

[30] Michael Seufert, Sebastian Egger, Martin Slanina, Thomas Zinner, Tobias Hoßfeld, and Phuoc Tran-Gia. 2014. A survey on quality of experience of HTTP adaptive streaming. *IEEE Communications Surveys & Tutorials* 17, 1 (2014), 469–492.

[31] Han Hee Song, Zihui Ge, Ajay Mahimkar, Jia Wang, Jennifer Yates, Yin Zhang, Andrea Basso, and Min Chen. 2011. Q-score: Proactive service quality assessment in a large IPTV system. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*. 195–208.

[32] Kaixin Sui, Mengyu Zhou, Dapeng Liu, Minghua Ma, Dan Pei, Youjian Zhao, Zimu Li, and Thomas Moscibroda. 2016. Characterizing and improving wifi latency in large-scale operational networks. In *Proceedings of the 14th Annual International Conference on Mobile Systems, Applications, and Services*. 347–360.

[33] Yi Sun, Xiaoqi Yin, Junchen Jiang, Vyas Sekar, Fuyuan Lin, Nanshu Wang, Tao Liu, and Bruno Sinopoli. 2016. CS2P: Improving video bitrate selection and adaptation with data-driven throughput prediction. In *Proceedings of the 2016 ACM SIGCOMM Conference*. 272–285.

[34] Mukundan Venkataraman and Mainak Chatterjee. 2011. Inferring video QoE in real time. *IEEE network* 25, 1 (2011), 4–13.

[35] Zheng Yang, Longfei Shangguan, Weixi Gu, Zimu Zhou, Chenshu Wu, and Yunhao Liu. 2014. Sherlock: Micro-environment sensing for smartphones. *IEEE Transactions on Parallel and Distributed Systems* 25, 12 (2014), 3295–3305.

[36] Chengwei Zhang, Angela Yunxian Wang, and Xiaojun Hei. 2017. Relay discovery and selection for large-scale P2P streaming. *Plos one* 12, 4 (2017), e0175360.

[37] Songyang Zhang. 2020. An Online Learning Based Path Selection for Multipath Video Telephony Service in Overlay. *arXiv preprint arXiv:2003.03575* (2020).

[38] Guowei Zhu, Chou Mo, Zhi Wang, and Wenwu Zhu. 2016. User Mapping Strategies in Multi-Cloud Streaming: A Data-Driven Approach. In *2016 IEEE Global Communications Conference (GLOBECOM)*. IEEE, 1–6.