P2P Live Streaming with Recommender Systems

Abhinau K. Venkataramanan, Joshua Ebenezer, Aniruddh Venkatakrishnan

April 2021

1 Introduction

1.1 Multi-channel P2P Live Streaming

This system involves people/devices live streaming various channel data(J) from different servers. The server bandwidth, in itself, is insufficient to satisfy the streaming-rate for all the users viewing the channel. To overcome this issue, each user also redistributes the data it receives into the system thereby increasing the effective bandwidth of each channel. Viewing and uploading are strictly decoupled in this model, which was introduced in [1]. All allocations are implemented with the aim of maximizing Probability of Universal Streaming (PUS) which gives the probability that all users of a channel get sufficient bandwidth for uninterrupted viewing.

1.2 User Distribution

The system dynamics involves two components: peer churn and channel churn. Channel churn refers to users jumping between each live stream, staying in each channel for a random duration (μ_j^{-1}). Peer churn refers to devices entering and exiting the system. This is true in the case of devices like laptops, PCs. Devices like TVs and set-top boxes can be considered as always on.
Since channel churn happens on a faster time scale as compared to peer churn, we first consider a closed system with a fixed number of users(n). Modelling this system as a closed Jackson Network with a transition probability matrix of $\mathbf{P}$, we get :

$$\boldsymbol{\lambda} = \mathbf{P^T}\boldsymbol{\lambda}$$

$$\rho_j = \lambda_j/\mu_j$$

$$\sum_{j=1}^{J}\rho_j = 1$$

$$P(M_1 = m_1, \ldots, M_J = m_J) = n!\frac{\rho_1^{m_1}}{m_1!}\cdots\frac{\rho_1^{m_J}}{m_J!}$$

Considering an open system with peer churn with arrival rate of $\boldsymbol{\nu}$ and departing rate of p_{i0}, we get :

$$\begin{aligned}
\boldsymbol{\lambda} &= \boldsymbol{\nu} + \boldsymbol{P^T}\left(\mathbf{I} - \mathbf{P_o}\right)\boldsymbol{\lambda} \\
\mathbf{P_o} &= \mathrm{diag}([p_{i0}]) \\
\rho_j &= \lambda_j/\mu_j
\end{aligned}$$

$$P(M_1 = m_1, ..M_J = m_J) = \prod_{j=1}^{J} \frac{\rho_j^{m_j}}{m_j!} e^{-\rho_j}$$

1.3 Universal Streaming

The resource index for channel j is defined by

$$\sigma_j(M_j) = \frac{b_j - o_j}{d_j(M_j)} \tag{1}$$

where b_j is the total upload bandwidth available for channel j, $d_j(M_j)$ is the total bandwidth required to achieve universal streaming in channel j given the number of viewers to be M_j, and o_j is VUD's bandwidth overhead for channel j. The probability of universal streaming for channel j is given by:

$$PU_j = P(\sigma_j(M_j) \geq 1) \tag{2}$$

The probability of system-wide universal streaming is given by:

$$PS = P(\sigma_j(M_j) \geq 1, j = 1 \ldots J) \tag{3}$$

1.4 Recommender Systems

Recommenders are used to shape the user's preferences, thereby *modifying* the channel churn matrix. They have been used, traditionally, in cached systems to decrease the latency of the system. We consider the possibility of tuning the TPM to obtain a user distribution which can increase the Universal Streaming probability. The optimal recommender must increase the PUS while ensuring that the recommendations are actually reasonable, by considering the similarity between the channels.

2 Load Balancing

The first approach we take towards finding an optimal recommender is to solve the problem in two steps. In the first step, we will compute an optimal load distribution $\boldsymbol{\rho^*}$ over the network. More concretely, we will solve the problem

$$\boldsymbol{\rho^*} = \arg\max PS(\boldsymbol{\rho}) \tag{4}$$

Then, we may employ several strategies to find the optimal recommender. For example, if we are only interested in achieving the optimal load distribution on a closed network, we solve the feasibility problem

$$
\begin{aligned}
\text{find} \quad & \mathbf{P} \\
\text{s.t.} \quad & \boldsymbol{\lambda}^* = \mathbf{P}^T \boldsymbol{\lambda}^* \\
& \lambda_i^* = \rho_i^* \mu_i \\
& P_{ii} = 0 \\
& P_{ij} \geq 0 \\
& \sum_j P_{ij} = 1
\end{aligned} \tag{5}
$$

While this is a linear feasibility problem, it may

- Lead to trivial solutions.
- Be infeasible.

The reason we obtain trivial solutions is that we accept any recommender that achieves the load distribution. In practice, however, users prefer recommenders that suggest content similar to the one they have just watched.

Assume a similarity matrix $\mathbf{S}$ where S_{ij} encodes the similarity between channels i and j. We can then search for recommenders that suggest similar content. This formulation will be discussed later. Our first focus will be on finding the optimal load distribution.

2.1 Optimal Load Balancing in a Closed Network

Consider the expression for system-wide universal streaming. Based on the system parameters (allocation of uploaders to channels and their upload rates), it can be shown that the condition for universal streaming corresponds to having less than a maximum number of customers in each channe. Let $\boldsymbol{\delta}$ be the vector of such maximum admissible users in each channel. We can then define the admissible region

$$
\mathcal{M}_{\boldsymbol{\delta}} = \{\mathbf{m} \mid \mathbf{0} \leq \mathbf{m} \leq \boldsymbol{\delta}, \mathbf{1}^T \mathbf{m} = n\} \tag{6}
$$

For simplicity, let $|\mathcal{M}_{\boldsymbol{\delta}}| = M_{\boldsymbol{\delta}}$ and $\mathcal{M}_{\boldsymbol{\delta}} = \{\mathbf{m_i} \mid i = 1 \dots M_{\boldsymbol{\delta}}\}$.

$$
PS = \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{m_{ij}} \tag{7}
$$

To find the optimal load distribution is to solve the following optimization problem.

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{m_{ij}} \\
\text{s.t.} \quad & \mathbf{1}^T \boldsymbol{\rho} = 1 \\
& \boldsymbol{\rho} \geq \mathbf{0}
\end{aligned} \tag{8}
$$

For now, let us ignore the constraint $\boldsymbol{\rho} \geq \mathbf{0}$. Then, we have an equality constrained maximization problem. This can be solved using the Lagrange multiplier method. The Lagrange function is given by

$$L(\boldsymbol{\rho}, \eta) = \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{m_{ij}} + \eta \left(\sum_{j=1}^{J} \rho_j - 1 \right) \tag{9}$$

The optimal solution is among the saddle points of the Lagrangian. It can be shown that $\boldsymbol{\rho^*}$ is a saddle point iff it satisfies

$$\rho_k^* = \frac{1}{\sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}}} \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}} \left(\frac{m_{ik}}{n} \right) \tag{10}$$

1. We can interpret the right-hand-side of the optimality condition in (10) as the average fraction of customers in channel k conditioned on being an admissible configuration, and the left-hand-side is the unconditional average fraction.
2. The condition in (10) is a fixed-point equation. Since the function is continuous and maps the J-dimensional probability simplex to itself, Brouwer fixed point theorem tells us that a solution exists. However, note that the solution may not be unique. In other words, (10) is a necessary condition for optimality, not sufficient.
3. We can use fixed-point iteration to solve for $\boldsymbol{\rho^*}$. This may work well in practice but we have not shown that the function is contractive. Therefore, the fixed-point iteration may not always converge.

Can we solve a (potentially) non-convex problem that guarantees optimality? Yes! To do so, we will use the result from Pascual et al. [2] to derive the following optimization that acts as a dual problem.

$$\begin{aligned} \min_{\mathbf{y}} \quad & -\sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \log \binom{n}{\mathbf{m_i}} + \sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \log y_i - \sum_{i=1}^{J} z_i \log z_i \\ \text{s.t.} \quad & \mathbf{z} = \mathbf{M}^T \mathbf{y} \\ & \mathbf{y} \geq 0 \\ & \mathbf{1}^T \mathbf{y} = 1 \end{aligned} \tag{11}$$

The optimal solution of the "dual" problem is related to the optimal solution of the primal as

$$\boldsymbol{\rho^*} = \frac{1}{n} \mathbf{z}^* \tag{12}$$

2.2 Optimal Load Balancing in an Open Network

Given a vector of maximum admissible users $\boldsymbol{\delta}$, the probability of system-wide universal streaming in an open network can be expressed as

$$PS = \sum_{\mathbf{0} \leq \mathbf{m} \leq \boldsymbol{\delta}} \prod_{j=1}^{J} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) = \prod_{j=1}^{J} \left\{ \sum_{0 \leq m_j \leq \delta_j} \frac{\rho_j}{m_j!} \exp(-\rho_j) \right\} \tag{13}$$

Then, the optimal load distribution is found by solving the following optimization problem

$$\begin{aligned} \max_{\boldsymbol{\rho}} \quad & \prod_{j=1}^{J} \left\{ \sum_{0 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \right\} \\ \text{s.t.} \quad & \boldsymbol{\rho} \ge \mathbf{0} \end{aligned} \tag{14}$$

Clearly, the product form of the distribution allows us to separate the problem in (14) into J problems of the form

$$\begin{aligned} \max_{\rho_j} \quad & \sum_{0 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \\ \text{s.t.} \quad & \rho_j \ge 0 \end{aligned} \tag{15}$$

Note that the first term of the summation corresponding to $m_j = 0$ becomes $\exp(-\rho_j)$, which is maximized by setting $\rho_j = 0$. This is an artefact of the fact that $\lim_{x\to 0} x^a \exp(-x) = 1(a = 0)$. To avoid this trivial solution, we will instead maximize the following lower bound to PU_j

$$\begin{aligned} \max_{\rho_j} \quad & \sum_{1 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \\ \text{s.t.} \quad & \rho_j \ge 0 \end{aligned} \tag{16}$$

Note that each term of the summation is quasi-concave, but the sum may not be quasiconcave. Since the constraint set of this problem is simple, we can solve this directly by setting the gradient to zero. Differentiating with respect to ρ_j,

$$\begin{aligned} \frac{dPU_j}{d\rho_j} &= \sum_{1 \le m_j \le \delta_j} \frac{m_j}{m_j!} \rho_j^{m_j - 1} \exp(-\rho_j) - \frac{1}{m_j!} \rho_j^{m_j} \exp(-\rho_j) \\ &= \exp(-\rho_j) \sum_{1 \le m_j \le \delta_j} \frac{1}{m_j!} \left(m_j \rho_j^{m_j - 1} - \rho_j^{m_j} \right) \\ &= \exp(-\rho_j) \sum_{1 \le m_j \le \delta_j} \left(\frac{\rho_j^{m_j - 1}}{(m_j - 1)!} - \frac{\rho_j^{m_j}}{m_j!} \right) \\ &= \exp(-\rho_j) \left(1 - \frac{\rho_j^{\delta_j}}{\delta_j!} \right) \end{aligned} \tag{17}$$

The first observation we can make is that PU_j is a quasiconcave function of ρ_j. Further, the optimal load (with respect to the lower-bound) is given by

$$\rho_j^* = \sqrt[\delta_j]{\delta_j!} \approx 0.3755\delta_j + 0.55 \tag{18}$$

Therefore,

$$\boldsymbol{\rho}^* \approx 0.3755 \cdot \boldsymbol{\delta} + 0.55 \tag{19}$$

We observe that the optimal load grows (approx.) linearly with the maximum admissible number of customers. However, it is important to note that because we have not included the recommender in this optimization problem, this load may not be achievable! Increasing $\boldsymbol{\delta}$ indefinitely will increases $\boldsymbol{\rho}^*$ indefinitely, while the flow into the system is fixed at $\boldsymbol{\nu}$. Similarly, decreasing $\boldsymbol{\delta}$ may decrease $\boldsymbol{\rho}^*$ below the minimum load $[\nu_j/\mu_j]$

2.3 Restricting the Domain to Achievable Loads

As we observed in the case of open network, the "optimal" load may not be achievable. Restricting the search space to achievable loads is equivalent to searching on the set

$$\{\boldsymbol{\rho} \mid \rho_j = \lambda_j/\mu_j, \exists \mathbf{P} \text{s.t.} \boldsymbol{\lambda} = \boldsymbol{\nu} + (\mathbf{I} - \mathbf{P_o})\,\mathbf{P}\boldsymbol{\lambda}\}.$$

Adding this as a constraint to the optimization problem in (14),

$$\begin{aligned}
\max_{\boldsymbol{\rho}} \quad & \prod_{j=1}^{J} \left\{ \sum_{1 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \right\} \\
\text{s.t.} \quad & \boldsymbol{\lambda} \ge \mathbf{0} \\
& \rho_j = \lambda_j/\mu_j \\
& \boldsymbol{\lambda} = \boldsymbol{\nu} + \mathbf{P}^T\,(\mathbf{I} - \mathbf{P_o})\,\boldsymbol{\lambda} \\
& P_{ii} = 0 \\
& P_{ij} \ge 0 \\
& \sum_j P_{ij} = 1 \qquad (20)
\end{aligned}$$

This problem is non-convex due to the presence of $\mathbf{P}^T\,(\mathbf{I} - \mathbf{P_o})\,\boldsymbol{\lambda}$, which is a bilinear term. Let $\mathbf{y} = \mathbf{P}^T\,(\mathbf{I} - \mathbf{P_o})\,\boldsymbol{\lambda}$. From the structure of $\mathbf{P}$, we see that the existence of $\mathbf{P}$ that if

$$\begin{aligned}
& y_i = \sum_{j \ne i} P_{ji}(1 - p_{jo})\lambda_j \qquad (21) \\
& P_{ii} = 0 \\
& P_{ij} \ge 0 \\
& \sum_j P_{ij} = 1 \qquad (22)
\end{aligned}$$

then

$$0 \le y_j \le \sum_{k \ne j} (1 - p_{ko})\lambda_k \qquad (23)$$

Therefore, we can eliminate $\mathbf{P}$ from the optimization problem, eliminating the bilinear constraint in the process, and write a relaxed version of the problem as

$$\begin{aligned}
\max_{\boldsymbol{\rho}} \quad & \prod_{j=1}^{J} \left\{ \sum_{1 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \right\} \\
\text{s.t.} \quad & \boldsymbol{\lambda} \ge \mathbf{0} \\
& \rho_j = \lambda_j/\mu_j \\
& \nu_j \le \lambda_j \le \nu_j + \sum_{k \ne j} (1 - p_{ko})\lambda_k, \quad \forall j = 1 \ldots J \qquad (24)
\end{aligned}$$

While this relaxation may not be tight, the constraint set is now a polyhedron. While each PU_j is quasi-concave, the product may not be quasi-concave in general. However, we can show a stronger result - the objective function is log-concave. Therefore, the problem in (24) involves maximizing a log-concave function over a convex set, which can be solved efficiently as a convex program.

3 Recommender objective for closed networks

As defined in Section 2, let $\mathbf{S}$ be the $J \times J$ channel similarity matrix, and let $\mathbf{P}$ be the recommendation matrix, i.e., P_{ij} is the transition probability from channel i to j as controlled by the recommender. We define the recommender objective as the average channel similarity observed by a user, i.e.,

$$S(\boldsymbol{\rho}, \mathbf{P}) = \sum_{i=1}^{J} \Pr(\text{customer in channel i}) \sum_{j=1}^{J} P_{ij} S_{ij} \tag{25}$$

$$= \sum_{i=1}^{J} \frac{\rho_i}{\sum_{k=1}^{J} \rho_k} \sum_{j=1}^{J} P_{ij} S_{ij} \tag{26}$$

$$= \sum_{i=1}^{J} \rho_i \sum_{j=1}^{J} P_{ij} S_{ij} \tag{27}$$

The load of each channel j, ρ_j, is defined as $\rho_j = \lambda_j / \mu_j$, where $\frac{1}{\mu_j}$ is the expected amount of time a peer continuously views channel j. λ is the vector that satisfies the flow conservation equations

$$\boldsymbol{\lambda} = \mathbf{P}^T \boldsymbol{\lambda} \tag{28}$$

This "constraint" on $\boldsymbol{\lambda}, \mathbf{P}$ is not linear or convex. However, with a change of variables

$$l_i = \log \lambda_i \tag{29}$$

$$y_{ij} = \log P_{ij}^T \tag{30}$$

the equation

$$\sum_j P_{ij}^T \lambda_j = \lambda_i \tag{31}$$

can be written as

$$\sum_{j, j \neq i} \exp\left(y_{ij} + l_j - l_i\right) = 1 \tag{32}$$

$$\log \left(\sum_{j \neq i} \exp\left(y_{ij} + l_j - l_i\right) \right) = 0 \tag{33}$$

for $i = 1 \ldots . J$. logsumexp is a convex function, but the constraint is an equality, not an inequality, and hence the feasible set is not convex.

Let $x_i = \log \rho_i$. Then the objective becomes

$$\max_{x,y} \quad \log\left(\sum_{i=1}^{J}\sum_{j=1,j\neq i}^{J} \exp\left(x_i + y_{ij} + \log S_{ij}\right)\right) \tag{34}$$

$$\text{s.t.} \quad \log\left(\sum_{j,j\neq i} \exp\left(y_{ij} + x_j - x_i - \log \mu_i + \log \mu_j\right)\right) = 0 \quad \forall i = 1 \ldots J$$

$$\sum_i \exp(x_i) = 1$$

$$\sum_i \exp(y_{ij}) = 1 \quad \forall j = 1 \ldots J$$

The first constraint is an equality over a convex function and will have to relaxed to make the feasible set convex, but we can show that relaxing the second and third constraint will not change the problem. The proof is shown in appendix D.

4 Asymptotic Analysis for Closed Networks

Our goal is to determine under what conditions PS is high for a system when the number of (uploading) peers approaches infinity. Let the uploading rate for each peer assigned to channel j be u_i^j, $i = 1 \ldots n_j$, r_j be the streaming rate of channel j, and v_j be the server rate for channel j. We have

$$b_j = v_j + \sum_{i=1}^{n_j} u_i^j \tag{35}$$

$$o_j = n_j r_j \tag{36}$$

$$d_j(M_j) = M_j r_j \tag{37}$$

4.1 Homogenous systems

We first consider the case when $u_i^j = u$ for all peers, with $u > r_j$ for $j = 1 \ldots .J$. Let $n_j = K_j n$, where K_j is a fraction such that $\sum_j K_j = 1$. As $n \to \infty$, we have

$$\begin{aligned}\lim_{n\to\infty} \sigma_j(M_j) &= \lim_{n\to\infty} \frac{b_j - o_j}{d_j(M_j)} \\ &= \lim_{n\to\infty} \frac{v_j + K_j n u - K_j n r_j}{M_j r_j} \\ &= \lim_{n\to\infty} \frac{K_j(u - r_j)}{\frac{M_j}{n} r_j}\end{aligned} \tag{38}$$

As $n \to \infty$, $\frac{M_j}{n} \to \rho_j$ (the load on the channel). Hence

$$\lim_{n\to\infty} \sigma_j(M_j) = \frac{K_j(u - r_j)}{\rho_j r_j} \tag{39}$$

$\sigma_j(M_j) \geq 1$ iff $\frac{K_j(u-r_j)}{\rho_j r_j} \geq 1$, which happens when

$$K_j \geq \frac{\rho_j r_j}{(u-r_j)} \tag{40}$$

Define

$$\alpha_j = \sum_j \frac{\rho_j r_j}{(u-r_j)} \tag{41}$$

If $\alpha_j \leq 1$, then for any $K_1, \ldots K_J$ with $K_j \geq \frac{\rho_j r_j}{(u-r_j)}$ and $\sum_j K_j = 1$, the probability of universal streaming (PS) goes to 1. In this case, we can define an optimization problem such that PS is 1 but the recommender objective is maximized, with the substitution of $\rho = \exp(x)$.

$$\begin{aligned} \max_{\mathbf{x},\mathbf{y}} \log & \left(\sum_{i=1}^{J} \sum_{j=1}^{J} \exp\left(x_i + y_{ij} + \log S_{ij}\right) \right) \qquad (42) \\ \text{s.t.} \ \log & \left(\sum_j \exp\left(y_{ij} + x_j - x_i + \log \mu_j - \log \mu_i\right) \right) = 0 \quad \forall i = 1 \ldots J \\ & \sum_i \exp(x_i) = 1 \\ & \sum_j \frac{\exp(x_j) r_j}{(u - r_j)} \leq 1 \\ & \sum_i \exp(y_{ij}) = 1 \quad \forall j = 1 \ldots J \end{aligned}$$

When the convex equalities are relaxed to inequalities, we can solve the relaxed problem as a convex problem.

4.2 Heterogeneous systems

We now consider the case where the upload rates of all the users are not the same. We consider two groups of peers with a separate upload rate for each. Let f be the fraction of low-upload rate peers, and $1-f$ be the fraction of high-upload rate peers. Let $n_j^l = K_j^l n$ be the fraction of low upload rate peers allocated to channel j, and $n_j^h = K_j^h n$ be the fraction of high upload rate peers allocated to channel j.

$$b_j = v_j + K_j^l n u^l + K_j^h n u^h \tag{43}$$

$$o_j = (K_j^l n + K_j^h n) r_j \tag{44}$$

$$\begin{aligned} \lim_{n\to\infty} \sigma_j(M_j) &= \lim_{n\to\infty} \frac{b_j - o_j}{d_j(M_j)} \qquad (45) \\ &= \lim_{n\to\infty} \frac{v_j + K_j^l n u^l + K_j^h n u^h - (K_j^l n + K_j^h n) r_j}{M_j r_j} \\ &= \frac{K_j^l (u^l - r_j) + K_j^h (u^h - r_j)}{\rho_j r_j} \end{aligned}$$

Therefore $\sigma_j(M_j) \geq 1$ iff $K_j^l(u^l - r_j) + K_j^l(u^h - r_j) \geq \rho_j r_j$ for all $j = 1 \ldots .J$. This can be heuristically converted to an optimization problem as

$$\max_{\mathbf{K^l},\mathbf{K^h},\boldsymbol{\rho}} \min_j y K_j^l(u^l - r_j) + K_j^h(u^h - r_j) - \rho_j r_j \tag{46}$$

One way to approach this problem is to consider it as a two-stage process. In the first stage, we optimize the load such that the allocator system is optimal, and in the second stage we optimize the recommender system under the constraint that the eigenvector of the transition probability matrix corresponds to the load obtained in the first stage.

$$\begin{aligned} \max_{\mathbf{K^h},\mathbf{K^l},\boldsymbol{\rho}} \quad & Z \\ \text{s.t.} \quad & Z \leq \min_j \xi_j K_j^l + \zeta_j K_j^h - r_j \rho_j \\ & \mathbf{1}^T \boldsymbol{\rho} = 1 \\ & \boldsymbol{\rho} \geq \mathbf{0} \\ & \mathbf{1}^T \mathbf{K^h} = 1 - f \\ & \mathbf{1}^T \mathbf{K^l} = f \end{aligned} \tag{47}$$

After we find the optimal $\boldsymbol{\rho}$ for this problem, we can then optimize the recommender using (34).

These stages are linear optimization problems by themselves and can easily be solved. If we aim to jointly optimize the recommender system and the allocation system, we have the following optimization

$$\begin{aligned} \max \quad & Z + \nu \log \left(\sum_{i=1}^J \sum_{j=1}^J \exp\left(x_i + y_{ij} + \log S_{ij}\right) \right) \\ \text{s.t.} \quad & Z \leq \min_j \xi_j K_j^l + \zeta_j K_j^h - r_j \rho_j \\ & \log \left(\sum_j \exp\left(y_{ij} + x_j - x_i + \log \mu_i - \log \mu_j\right) \right) = 0 \quad \forall i = 1 \ldots J \\ & \sum_i \exp(x_i) = 1 \\ & \sum_i \exp(y_{ij}) = 1 \text{ for } j = 1 \ldots J \\ & \mathbf{1}^T \mathbf{K^h} = 1 - f \\ & \mathbf{1}^T \mathbf{K^l} = f \end{aligned} \tag{48}$$

Once again, relaxing the logsumexp constraint and the constraint on the $1^T \exp(x)$ reduces this to a convex program.

5 Recommender systems for Open Networks

The primary objective for recommenders in an open network is now slightly different because $\boldsymbol{\rho}$ is no longer normalized to 1, i.e.

$$\max_{\boldsymbol{\rho},\mathbf{P}} \sum_{i=1}^J \sum_{j=1}^J \frac{\rho_i}{\sum_{k=1}^J \rho_k} P_{ij} S_{ij} \tag{49}$$

To deal with this, we introduce another variable

$$\tilde{\boldsymbol{\rho}} = \frac{\boldsymbol{\rho}}{\sum_{k=1}^{J} \rho_k} \tag{50}$$

For the both the homogenous and the heterogenous case $\frac{M_j}{n} \to \tilde{\rho_j}$, and hence $\tilde{\boldsymbol{\rho}}$ can directly be used in the constraint equations in place of $\boldsymbol{\rho}$. The constraints on the load vector ρ and the arrival rate vector λ are different. Specifically,

$$\boldsymbol{\lambda} = \boldsymbol{\nu} + \mathbf{P^T}(\mathbf{I} - \mathbf{P_0})\boldsymbol{\lambda} \tag{51}$$

where P is the channel churn matrix and ν_i is the exogenous arrival rate for channel i, and $\mathbf{P_0} = \text{diag}([p_{i0}])$ where p_{i0} is the exit probability from state i. p_{i0} and $\boldsymbol{\nu}$ are known quantities. The constraint can be written elementwise as

$$\lambda_i = \nu_i + \sum_{j \neq i} P_{ji}(1 - p_{j0})\lambda_j \tag{52}$$

Substituting $l_i = \log(\lambda_i)$ and $y_{ij} = \log(P_{ji})$ gives

$$\log\left(\exp\left(\log \nu_i - l_i\right) + \sum_{j \neq i} \exp\left(y_{ij} + l_j + \log(1 - p_{j0})\right)\right) = 0 \tag{53}$$

This is a logsumexp expression, similar to what we obtained in the closed network case.

6 Numerical results

6.1 Fixed Point Equation in Closed networks

In this evaluation, we consider 20 users, 6 channels. The maximum limit on users for each channel is given by [9,6,5,9,7,8] respectively. We start with an initial distribution having all the load at the first channel. We show a plot on the variation of the objective function value with the iteration count.

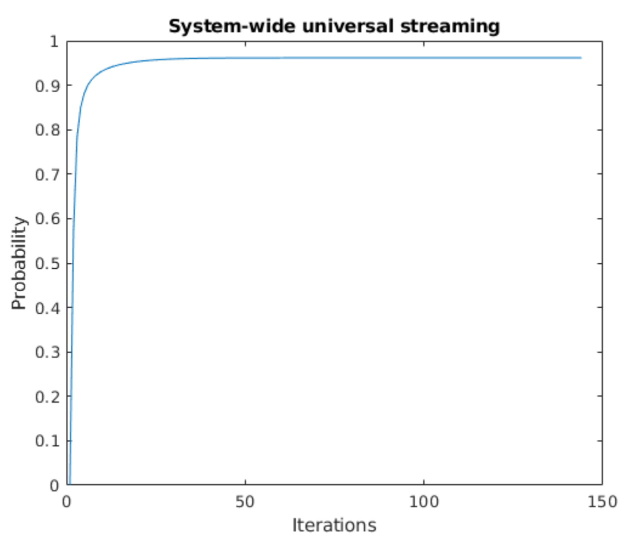

Figure 1: Variation of Objective value

6.2 Relaxed Solution to the Open Network

In this evaluation we compare the results obtained by the domain restricted problem and it's relaxed version. For this, we considered a system with 6 channels, an arrival rate of [0.03,0.03,0.03,0.03,1e-3,0.03], a mean staying time of [0.54,0.55,1,0.52,0.55,0.51], and a maximum user limit of [9,6,5,9,7,8].
Running the numerical algorithm shows that the obtained objective value of the relaxed version is within 8% of the actual solution, and the obtained user load is within 11%. This shows that we can use the relaxed problem formulation to get a close enough solution at much lower computational complexity.

6.3 Asymptotic homogeneous systems

Here we compare the results between joint optimization of PUS and similarity, and step-wise optimization of the load and then the similarity matrix. The objective value is a measure of correlation between the recommender and the similarity matrix. We consider a 6 channel scenario with channel rates approximately half of the upload rate.

Channel Rates	Similarity matrix	Joint Objective	Step-wise Objective
R1	S1	1.75	1.51
R1	S2	1.67	1.16
R1	S3	1.19	1.64
R1	S4	1.88	1.63
R1	S5	0.82	1.79
R2	S1	0.81	1.30
R2	S2	1.09	1.41
R2	S3	1.57	1.07
R2	S4	1.74	0.52
R2	S5	1.40	1.50
R3	S1	1.68	1.30
R3	S2	2.36	1.41
R3	S3	1.95	1.07
R3	S4	1.77	0.52
R3	S5	1.10	1.50

While it look possible that sometimes the step-wise solver can give a better correlation, the load distribution obtained from the step-wise optimizer is an edge-case where all the load is on a single channel which has the lowest streaming rate. This scenario will not happen in practice.

References

[1] Di Wu, Yong Liu, and Keith Ross. Queuing network models for multi-channel P2P live streaming systems. In *IEEE INFOCOM 2009*, pages 73–81, 2009.

[2] Luis D. Pascual and Adi Ben-Israel. Constrained maximization of posynomials by geometric programming. *Journal of Optimization Theory and Applications*, 5(2):73–80, Mar 1970.

Appendix A Deriving the Fixed Point Condition

Recall that the Lagrange function is given by

$$L(\boldsymbol{\rho}, \eta) = \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{m_{ij}} + \eta \left(\sum_{j=1}^{J} \rho_j - 1 \right) \tag{54}$$

The optimal solution is among the saddle points of the Lagrangian. To find the saddle point, we will set the derivatives to zero. Computing the derivatives,

$$\frac{\partial L}{\partial \rho_k} = \eta + \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{m_{ij}} m_{ik} \rho_k^{-1} \tag{55}$$

Therefore, at a saddle point $\boldsymbol{\rho^*}, \eta^*$,

$$\eta^* + \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}} m_{ik} \rho_k^{*-1} = 0 \tag{56}$$

$$\rho_k^* = -\sum_{i=1}^{M_{\boldsymbol{\delta}}} \frac{m_{ik}}{\eta^*} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}} \tag{57}$$

Enforcing the constraint that ρ_k^* sum to 1, we get

$$\rho_k^* = \frac{\sum_{i=1}^{M_{\boldsymbol{\delta}}} m_{ik} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}}}{\sum_{k=1}^{J} m_{ik} \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}}} \tag{58}$$

$$= \frac{1}{\sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}}} \sum_{i=1}^{M_{\boldsymbol{\delta}}} \binom{n}{\mathbf{m_i}} \prod_{j=1}^{J} \rho_j^{*m_{ij}} \left(\frac{m_{ik}}{n} \right) \tag{59}$$

Appendix B Deriving the Dual-like Problem for Universal Streaming in Closed Networks

As mentioned earlier, we will use the result from Pascual et al. [2].

Consider the maximization problem

$$\begin{aligned} \max \quad & g_0(\mathbf{x}) \\ \text{s.t.} \quad & g_k(\mathbf{x}) \le 1, \quad k = 1 \dots p \\ & x_j > 0, \quad j = 1 \dots d \end{aligned} \tag{60}$$

where

$$g_k(\mathbf{x}) = \sum_{i \in J[k]} c_i \prod_{j=1}^{d} x_j^{a_{ij}} \tag{61}$$

where

$$J[k] = \left\{ \sum_{l<k} n_l + 1, \sum_{l<k} n_l + 2, \sum_{l<k} n_l + n_k \right\}. \tag{62}$$

The following min-max problem acts as a "dual" program to (60).

$$\begin{aligned}
\min_{\mathbf{y}} \max_{\mathbf{z}} \quad \nu(\mathbf{y}, \mathbf{z}) = & \prod_{i=1}^{n_0} (c_i/y_i)^{-y_i} \prod_{i>n_0} (c_i/z_i)^{z_i} \prod_{k=1}^{p} S_k(\mathbf{z})^{S_k(\mathbf{z})} \\
\text{s.t.} \quad & -\sum_{i=1}^{n_0} y_i a_{ij} + \sum_{i>n_0} z_i a_{ij} = 0, \quad j = 1 \dots d \\
& z_i \geq 0, \quad i \in J[k], \quad k = 1 \dots p \\
& \mathbf{y} \geq 0 \\
& \mathbf{1}^T \mathbf{y} = 1
\end{aligned} \tag{63}$$

where

$$S_k(\mathbf{z}) = \sum_{i \in J[k]} z_i \tag{64}$$

Note: A posynomial optimization problem is said to be **superconsistent** if there exists a feasible point that satisfies all inequality constraints strictly. If the maximization problem in (60) is superconsistent, then the minmax problem in (63) is "tight", i.e.,

$$\{\max_{\mathbf{x}} g_0(\mathbf{x})\}^{-1} = g_0(\mathbf{x}^*)^{-1} = \inf_{\mathbf{y}} \sup_{\mathbf{z}} \nu(\mathbf{y}, \mathbf{z}) = \nu(\mathbf{y}^*, \mathbf{z}^*) \tag{65}$$

and the optimal points satisfy the relationship

$$c_i \prod_{j=1}^{d} \left(x_j^*\right)^{a_{ij}} = \begin{cases} y_i^*/\nu(\mathbf{y}^*, \mathbf{z}^*) & i \in J[0] \\ y_i^*/S_k(\mathbf{y}^*) & i \in J[k], \quad k = 1 \dots p \end{cases} \tag{66}$$

Converting a nice log-sum-exp to such a min-max problem may not seem appealing at first. However, in our case, we will see that this min-max problem simplifies greatly. We begin by making the following observations by comparing our maximization problem to the general one in (60),

1. $p = 1$
2. $d = J$
3. $n_0 = M_{\boldsymbol{\delta}}$
4. $c_i = \begin{cases} \log \binom{n}{\mathbf{m_i}} & i = 1 \dots M_{\boldsymbol{\delta}} \\ 1 & i = M_{\boldsymbol{\delta}} + 1 \dots M_{\boldsymbol{\delta}} + J \end{cases}$
5. Denoting $\mathbf{A} = [a_{ij}]$ and $\mathbf{M} = [m_{ij}]$, $A = \begin{bmatrix} \mathbf{M} \\ \mathbf{I}_J \end{bmatrix}$, where $\mathbf{A} \in \mathbb{R}^{(M_{\boldsymbol{\delta}}+J) \times J}, \mathbf{M} \in \mathbb{R}^{M_{\boldsymbol{\delta}} \times J}$, and $\mathbf{I}_J$ denotes the $J \times J$ identity matrix.

From item 5 in the list above, we observe that $\mathbf{A}$ has full column rank. Therefore, the linear equality constraint of the min-max problem in (63) has a unique solution. Substituting $\mathbf{A}$, the linear equality constraint simplifies to

$$\mathbf{z} = \mathbf{M}^T \mathbf{y} \tag{67}$$

Since $\mathbf{y} \geq \mathbf{0}$ and $m_{ij} \geq 0$, $\mathbf{z} \geq \mathbf{0}$. Therefore, $\mathbf{M}^T\mathbf{y}$ is feasible and the constraint set of the maximization part reduces to one point. Therefore, the dual reduces to the following minimization problem.

$$\begin{aligned} \min_{\mathbf{y}} \quad & \prod_{i=1}^{M_{\mathbf{z}}} (c_i/y_i)^{-y_i} \prod_{i=1}^{J} (1/z_i)^{z_i} \left(\mathbf{1}^T\mathbf{z}\right)^{\mathbf{1}^T\mathbf{z}} \\ \text{s.t.} \quad & \mathbf{z} = \mathbf{M}^T\mathbf{y} \\ & \mathbf{y} \geq 0 \\ & \mathbf{1}^T\mathbf{y} = 1 \end{aligned} \tag{68}$$

We make one final observation to simplify the problem further.

$$\mathbf{1}^T\mathbf{z} = \mathbf{1}^T \left(\sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \mathbf{m_i} \right) = \sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \left(\mathbf{1}^T \mathbf{m_i}\right) = n \sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i = n \tag{69}$$

This eliminates an exponent term from (68) leading to the equivalent minimization problem

$$\begin{aligned} \min_{\mathbf{y}} \quad & \prod_{i=1}^{M_{\boldsymbol{\delta}}} (c_i/y_i)^{-y_i} \prod_{i=1}^{J} (1/z_i)^{z_i} \\ \text{s.t.} \quad & \mathbf{z} = \mathbf{M}^T\mathbf{y} \\ & \mathbf{y} \geq 0 \\ & \mathbf{1}^T\mathbf{y} = 1 \end{aligned} \tag{70}$$

Equivalently, we can minimize the log of the objective, i.e.,

$$\begin{aligned} \min_{\mathbf{y}} \quad & -\sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \log \binom{n}{\mathbf{m_i}} + \sum_{i=1}^{M_{\boldsymbol{\delta}}} y_i \log y_i - \sum_{i=1}^{J} z_i \log z_i \\ \text{s.t.} \quad & \mathbf{z} = \mathbf{M}^T\mathbf{y} \\ & \mathbf{y} \geq 0 \\ & \mathbf{1}^T\mathbf{y} = 1 \end{aligned} \tag{71}$$

Clearly, our primal problem (8) is superconsistent, since $(1/J)\mathbf{1}$ is feasible. Therefore, the dual problem is tight, and we can use the relationship in (66) to relate the optimal values. Since $p = 1$, let us choose $k = 1$, which leads to $c_i = 1$ and $a_{ij} = 1(i - M_{\boldsymbol{\delta}} = j)$. Using the fact that $\mathbf{1}^T\mathbf{z} = n$,

$$\boldsymbol{\rho}^* = \frac{1}{n}\mathbf{z}^* \tag{72}$$

Appendix C Deriving Log-Concavity of the Universal Streaming Objective for Open Networks

To prove this, we first show that each term of the product is log-concave. Let $g_j(\rho_j) = \log PU_j(\rho_j)$. Then,

$$\begin{aligned} g_j'(\rho_j) &= \frac{PU_j'(\rho_j)}{PU_j(\rho_j)} \\ &= \frac{1}{PU_j(\rho_j)} \sum_{1 \le m_j \le \delta_j} \left(\frac{\rho_j^{m_j-1}}{(m_j-1)!} \exp(-\rho_j) - \frac{\rho_j^{m_j}}{m_j!} \exp(-\rho_j) \right) \\ &= \frac{1 - \frac{\rho_j^{\delta_j}}{\delta_j!}}{\sum_{1 \le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!}} \end{aligned} \tag{73}$$

Therefore,

$$g_j''(\rho_j) = \frac{\left(\sum_{1\le m_j \le \delta_j} \frac{\rho_j^{m_j}}{m_j!}\right)\left(-\frac{\rho_j^{\delta_j-1}}{(\delta_j-1)!}\right) - \left(1 - \frac{\rho_j^{\delta_j}}{\delta_j!}\right)\left(\sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j-1}}{(m_j-1)!}\right)}{\left(\sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j}}{m_j!}\right)^2} \tag{74}$$

To show that g is concave, we must check

$$\left(\sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j}}{m_j!}\right) \frac{\rho_j^{\delta_j-1}}{(\delta_j-1)!} + \left(1 - \frac{\rho_j^{\delta_j}}{\delta_j!}\right) \sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j-1}}{(m_j-1)!} \overset{?}{\ge} 0 \tag{75}$$

$$\sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j-1}}{(m_j-1)!} + \sum_{1\le m_j\le\delta_j} \rho_j^{m_j+\delta_j-1} \left(\frac{1}{m_j!(\delta_j-1)!} - \frac{1}{(m_j-1)!\delta_j!}\right) \overset{?}{\ge} 0 \tag{76}$$

$$\sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j-1}}{(m_j-1)!} + \sum_{1\le m_j\le\delta_j} \frac{\rho_j^{m_j+\delta_j-1}}{(m_j-1)!(\delta_j-1)!} \left(\frac{1}{m_j} - \frac{1}{\delta_j}\right) \overset{?}{\ge} 0 \tag{77}$$

Since $m_j \le \delta_j$, $1/m_j \ge 1/\delta_j$. Therefore, $g_j''(\rho_j) \le 0$, i.e., g_j is indeed concave. Further, the objective function is a product of log-concave functions - therefore it is also log-concave.

Appendix D Relaxing the recommender problem

Consider the substitution $\exp(z) = \alpha \exp(x)$, where $0 \le \alpha \le 1$. Substituting this in the second constraint will not change it because

$$\log\left(\sum_{j,j\ne i} \exp\left(y_{ij} + x_j - x_i - \log\mu_i + \log\mu_j\right)\right) = \log\left(\sum_{j,j\ne i} \exp\left(y_{ij} + z_j - z_i - \log\mu_i + \log\mu_j\right)\right) \tag{78}$$

Substituting $\exp(z_i) = \alpha \exp(x_i)$ in the objective function gives

$$\log\left(\sum_{i=1}^{J} \sum_{j=1, j\neq i}^{J} \exp\left(z_i + y_{ij} + \log S_{ij}\right)\right) = \log\left(\sum_{i=1}^{J} \sum_{j=1, j\neq i}^{J} \exp\left(x_i + y_{ij} + \log S_{ij}\right)\right) + \log\alpha \tag{79}$$

This means that increasing α increases the value of the objective function and hence the solution under the constraint $\sum_i \exp(x_i) \leq 1$ is equal to the solution under the constraint $\sum_i \exp(x_i) = 1$. In general, the first constraint will have to be relaxed to make the problem convex. If the first constraint is relaxed, the third constraint ($\sum_i \exp(y_{ij}) = 1 \quad \forall j = 1 \ldots J$) can be relaxed without changing the value of the optimization objective, using the same argument as before for the second constraint. This formulation will turn out to be useful for subsequent objectives.