A robust P2P streaming architecture and its application to a high quality live-video service

Héctor Cancela [a], Franco Robledo Amoza [a],
Pablo Rodríguez-Bocca [a,b,1], Gerardo Rubino [b] and
Ariel Sabiguero [a,b]

[a] *Instituto de Computación, Universidad de la República, Montevideo, Uruguay.*
[b] *IRISA - INRIA, Rennes, France.*

Abstract

This paper explores the design of a P2P architecture that sends real-time video over the Internet. The aim is to provide good quality levels in a highly dynamic P2P topology, where the frequent connections/disconnections of the nodes makes it difficult to offer the Quality-of-Experience (QoE) needed by the client.

We study a multi-source streaming approach where the stream is decomposed into several flows sent by different peers to each client including some level of redundancy, in order to cope with the fluctuations in network connectivity. We employ the recently proposed PSQA technology for evaluating automatically the perceived quality at the client side. We introduce a mathematical programming model to maximize the global expected QoE of the network (evaluated using PSQA), selecting a P2P connection scheme which enhances topology robustness. In addition, we provide an approximated algorithm to solve the proposed model, and we apply it to solve a case study based on real life data.

[1] Email: `prbocca@fing.edu.uy`

1 P2P Robust Assignment Model

This work studies the characteristics of a P2P based solution for live video broadcasting. In particular, in this section we develop a mathematical programming model for the stream assignment problem in order to maximize the expected perceived quality taking into account the network dynamics.

Time. The system is reconfigurated at discrete points in time, every Δt. Let us use Δt as unit of time, and denote by I_t the tth interval $(t, t+1]$.

Distribution scheme. The stream is originally sent to the clients (or terminals) by a brodcaster node s. Some clients act also as servers, relaying streams to other clients. For this purpose, each node v has an available output bandwidth BW_v^{out}. The system distributes a single stream of live video by means of K sub-streams denoted by $\sigma_1, \sigma_2, ..., \sigma_K$. Each substream σ_k is sent with a constant bandwidth bw_k. The total bandwidth of the stream is $\sum_{k=1}^{K} bw_k$. When a client receives the K substreams, it reproduces perfectly the stream. If it does not receive all the K substreams, it will reproduce a stream that can have a lower quality, depending on which substreams it receives and which redundant scheme is used (not discussed here).

Dynamics. The evolution of the system from t to $t+1$ is as follows: some nodes leave in I_t, possibly disconnecting other clients in some substreams; at the same time, some nodes enter the network requesting for connection; they remain isolated from the rest of the nodes until $t+1$ when the new reconfiguration action is performed. The goal of the reconfiguration is to reconnect the disconnected nodes and to connect the new arrivals to the network. The connexion scheme always builds trees of peers. At time $(t+1)^-$, just before the reconfiguration, the general situation is the following. For each substream σ_k there is a *principal tree*, $\mathcal{P}_k$, containing (at least) the source s; all its nodes receive substream σ_k. There are also $M_k \geq 0$ other trees, disjoint between them and with $\mathcal{P}_k$, denoted by $\tau_{k,1}, \tau_{k,2}, \cdots, \tau_{k,M_k}$; their nodes do not receive σ_k. The set of trees associated with substream σ_k is called *plane* k. A *perfect network* is a system where for each substream σ_k there is only one directed tree ($\mathcal{P}_k$, the principal one), meaning that $M_k = 0$. One of the goals of the reconfiguration action is to build a perfect network.

Quality. To evaluate the quality at the client side, we use the PSQA technology. Pseudo-Subjective Quality Assessment (PSQA) [3] is a general procedure that automatically measures the perceived quality, accurately and in real time. For instance, if we assume that the PSQA metric is scaled on [0..1] and if the network is perfect, its instantaneous total quality is equal to the number of connected clients (because $PSQA = 1$ for each client).

Optimization. As said before, the reconfiguration action will try to build a perfect network, and, among the huge number of possible perfect newtorks, it will try to build a robust one. For this purpose, we keep statistics about the nodes' behavior allowing us to evaluate their expected departure times from the network. Specifically, we maintain an estimate p_i of the probability that node i remains connected in the current period, when it is connected at the beginning (see below).

Formal Model. We propose now an Integer Mathematical Programming Model which contemplates the P2P dynamics (described above) in a time interval $(t, t+1]$. Consider the system at time t^+ and let $\mathcal{N}(t)$ be the set of connected nodes at t^+ ($N = \|\mathcal{N}(t)\|$). Define for each $k \in \{1, 2, \cdots, K\}$ and $i, j \in \mathcal{N}(t)$,

$$x_{i,j}^k = \begin{cases} 1 & \text{if node } i \text{ sends } \sigma_k \text{ to node } j, \\ 0 & \text{otherwise,} \end{cases}$$

$$y_{i,j}^k = \begin{cases} 1 & \text{if node } i \text{ precedes node } j \text{ in the tree containing } j \text{ in plane } k, \\ 0 & \text{otherwise.} \end{cases}$$

Since the perceived quality at node i depends on which substream is received by i, we assume available a function $f()$ of K variables such that the quality at time t experienced by node $i \in \mathcal{N}(t)$ (and measured using PSQA) is $Q_i = f(y_{s,i}^1, y_{s,i}^2, .., y_{s,i}^K)$.

For all $i \in \mathcal{N}(t)$, let z_i be the binary random variable equal to 1 if node i remains connected until $t+1$, 0 otherwise, where $\Pr(z_i = 1) = p_i$. The sets $\{x_{ij}^k\}$ and $\{y_{ij}^k\}$ specify the network configuration at t whereas the network configuration at $t+1$ is determined by the variables $\{\tilde{x}_{ij}^k\}$ and $\{\tilde{y}_{ij}^k\}$ satisfying the following relations: $\tilde{x}_{i,j}^k = z_i x_{i,j}^k z_j$, $\tilde{y}_{i,j}^k = \tilde{x}_{i,j}^k + \sum_{l=1}^N z_i \tilde{y}_{i,l}^k \tilde{x}_{l,j}^k$, and $\tilde{y}_{i,j}^k \leq y_{i,j}^k$.

The PSQA evaluation of the quality as perceived by node i at time $t+1$ is a random variable which is a function of $\{\tilde{y}_{s,i}^k\}$ and r.v. $\{z_i\}$. We will maximize its expected value, $\mathrm{E}\{\tilde{Q}_i\}$, where $\tilde{Q}_i = f(\tilde{y}_{s,i}^1, \tilde{y}_{s,i}^2, .., \tilde{y}_{s,i}^K)$ (actually, we use a scaled expression, see Fig. 1, which shows the complete model).

2 Algorithmic solution based on GRASP

GRASP [4] is a well known metaheuristic that has been successfully used to solve many hard combinatorial optimization problems. It is an iterative process which operates in two phases. In the *Construction Phase* an initial feasible solution is built whose neighborhood is then explored in the *Local Search Phase*. Next, we present a GRASP customized to solve our problem.

$$\max \mathsf{E}\left\{\frac{\sum_{i=1}^{N}\tilde{Q}_i}{\sum_{i=1}^{N} z_i}\right\} \quad \text{// global expected perceived quality}$$

st:

$$y_{i,j}^k + y_{j,i}^k \leq 1, \forall i,j \in N, \forall k \in K, \quad \text{// loops are not allowed}$$

$$y_{i,j}^k = x_{i,j}^k + \sum_{l=1}^{N} y_{i,l}^k x_{l,j}^k, \forall i \in N, \forall j \in N - \{s\}, \forall k \in K, \quad \text{// precedence constraints}$$

$$\sum_{i=1}^{N} x_{i,j}^k \leq 1, \forall j \in N, \forall k \in K, \quad \text{// each substream arrives from an only node}$$

$$\sum_{k=1}^{K}\{bw^k \sum_{j=1}^{N} x_{i,j}^k\} \leq BW^{out}(i), \forall i \in N, \quad \text{// bandwidth capacity constraints}$$

$$x_{i,j}^k = 1, \forall i \in N, \forall j \in N, \forall k \in K | x_{i,j}^k \in E_t^k, \quad \text{// network configuration at } t$$

$$\tilde{x}_{i,j}^k = z_i x_{i,j}^k z_j, \forall i \in N, \forall j \in N, \forall k \in K, \quad \text{// a link is preserved if source and terminal do not leave}$$

$$\tilde{y}_{i,j}^k = \tilde{x}_{i,j}^k + \sum_{l=1}^{N} z_i \tilde{y}_{i,l}^k \tilde{x}_{l,j}^k, \forall i \in N, \forall j \in N - \{s\}, \forall k \in K, \quad \text{// } \tilde{y} \text{ represents the precedence at } t+1$$

$$\tilde{y}_{i,j}^k \leq y_{i,j}^k, \forall i \in N, \forall j \in N, \forall k \in K,$$

$$z_i \sim Bern(p_i), \forall i \in N, \quad \text{// random variables with Bernoulli distribution (parameters } p_i\text{)}$$

$$x_{i,j}^k, y_{i,j}^k \in \{0,1\}, \forall i \in N, \forall j \in N, \forall k \in K$$

Fig. 1. Mathematical Programming Model.

Construction phase. Let's assume that the initial graph is not connected, and that the distribution trees were pruned in some arcs. We will identify disconnected subtrees $\tau_{k,i}$ by their roots for each substream k and nodes i with available bandwidth. The tree that contains the stream source will be called the main tree $\mathcal{P}_k$, for each substream k. Let p_a_list be the set of all possible assignments that can be performed in that state, which reconnect the graph. Each component of the assignment is a triplet (source node, target node, substream), and belongs to the set if and only if: (i) the source node belongs to the main tree, (ii) the target node is root of some disconnected tree, (iii) the source node has enough bandwidth to transmit the substream. Fig. 2 describes the construction of the initial solution. The procedure constructively builds the initial solution starting from the disconnected graph. A set, RCL, of the eligible assignments which result in the larger improvements to the current partial solution is constructed. The parameter n determines the size of the set RCL. At each iteration one of the elements from RCL is added to the current solution, this decision is made determining randomly the element $current\_a$ from the set RCL. The set p_a_list has to be updated due to the assignation of $current\_a$. The p_a_list set may become empty if no disconnected trees

```
Procedure GRASP_P2P_Network;
Input: G, k;

a_list ← empty_list;
current_psqa ← 0; best_psqa ← −1;
for i = 1 to MAXITER do
   [𝒢_sol, a_list] ← P2P_Const_Phase(G, a_list);
   [𝒢_sol, a_list] ← P2P_Local_Search(𝒢_sol, k);
   current_psqa ← PSQA_Expected(𝒢_sol) ;
   if (current_psqa > best_psqa) then
      best_psqa ← current_psqa;
      𝒢_best ← 𝒢_sol;
   end_if;
   a_list ← empty_list;
end_for
return 𝒢_best;
```

```
Procedure P2P_Const_Phase;
Input: G; a_list, n;

p_a_list ← Possible_Assignments(G) ;
while not_empty(p_a_list) do
   RCL ← Compute_RCL(p_a_list, n);
   current_a ← SelectRandom(RCL);
   a_list ← Add_Assignment(a_list, current_a);
   p_a_list ← Update_PA(G, current_a, p_a_list);
end_while;
𝒢_sol ← Apply_Assignments(G, a_list);
return 𝒢_sol, a_list;
```

Fig. 2. Customized GRASP and Construction Phase.

remain. It might be also the case that there is no more available bandwidth in any node of the main tree, thus, no allocation is possible and p_a_list becomes empty. These cases are the stop conditions.

Local search phase. The previous procedure generates a random solution a_list. To improve the solution constructed, a local search is applied. Following directly the ideas in [1], we use a RNN in the local search phase. After the m executions of the algorithm, $\hat{m}$ different initial solutions are obtained: $a\_list_1, a\_list_2, \ldots, a\_list_{\hat{m}}$.

Metric for enhancement of a given assignment. Selecting one solution amongst the $\hat{m}$ constructed is done by a Monte Carlo estimation of their future PSQA (by randomly simulating sequences of nodes entries and exits). The selected solution will be the one with higher mean PSQA.

3 Numerical Results and Discussion

We compare the performance of our P2P assignment metaheuristic with a traditional Content Delivery Network (CDN) [2] based implementation. Two important variables have to be considered in the analysis: the global perceived quality of the network, and the total bandwidth consumption (at the broadcaster and at the peers). We have statistical data coming from a live video delivery service of a medium size ISP, with 10.000 access of different users per month, and an average of 100 concurrent users per live-TV channel.

In the CDN architecture case, the broadcaster is a set of servers in the ISP datacenter, where the bandwidth consumption is the most expensive component cost of the service. The broadcaster absorbs all the load of the clients, with a stream of 512 Kbps; this means at least 50 Mbps of bandwidth in peak use. The broadcaster of the CDN study case has no failures in the service. If we consider the packet losses in the network negligible, we can assume that the CDN network has a perfect global quality (i.e $Q = 1.00$). We simulate a P2P architecture with the same clients' behavior (connections/disconnections) than the real CDN. Our results show that the broadcaster absorbs only 5.6 Mbps, and the peers the rest of the load (in average 0.6 Mbps per peer). The quality is not considerably degraded, with a worst case of $Q = 0.966$ on the average.

As main conclusions, we think that the numerical results obtained show the interest of employing a live-video P2P distribution system following a multi-source procedure. The PSQA technique allows to automatically measure the perceived quality as seen by the final users, and this in turn is a key issue in developing a formalized model and algorithmic solution procedures to define the topology and flow assignment of such a system.

References

[1] Cancela, H., F. Robledo and G. Rubino, *A GRASP algorithm with RNN based local search for designing a WAN access network*, Electronic Notes on Discrete Mathematics **18C** (2004), pp. 53–58.

[2] Cranor, C., M. Green, C. Kalmanek, D. Shur, S. Sibal and J. V. der Merwe, *Enhanced streaming services in a content distribution network*, IEEE Internet Computing (2001), pp. 66–75.

[3] Mohamed, S. and G. Rubino, *A Study of Real–time Packet Video Quality Using Random Neural Networks*, IEEE Transactions On Circuits and Systems for Video Technology **12** (2002), pp. 1071 –1083.

[4] Resende, M. G. C. and C. C. Ribeiro, "Greedy Randomized Adaptive Search Procedures," In F. Glover and G. Kochenberger, editors, Handbook of Methaheuristics, Kluwer Academic Publishers, 2003, 219-249 pp.