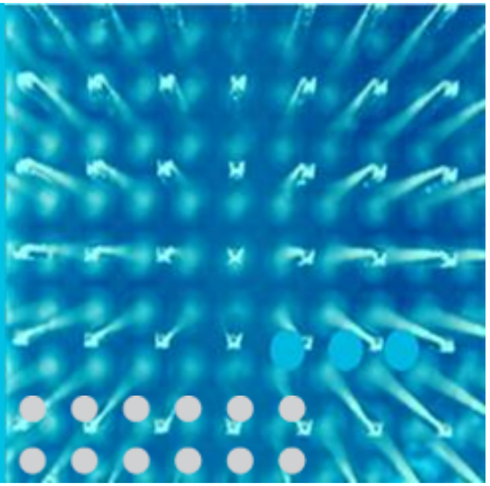
Alcatel·Lucent

On Server Dimensioning for Hybrid Peer-to-Peer Content Distribution Networks



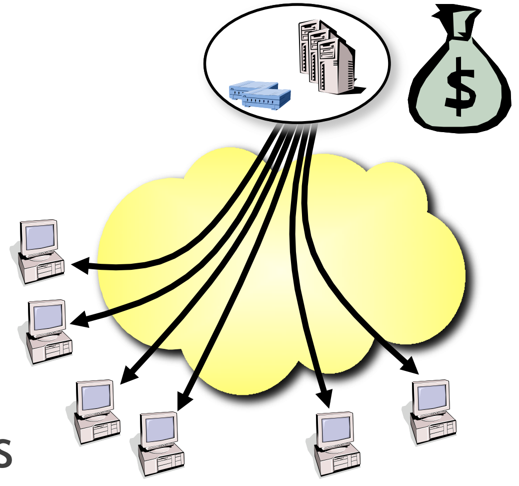
P2P’08

Ivica Rimac, Anwar Elwalid, Sem Borst

Bell Laboratories

September 2008

Motivation
Cost-Efficient and Reliable Content Distribution Solutions

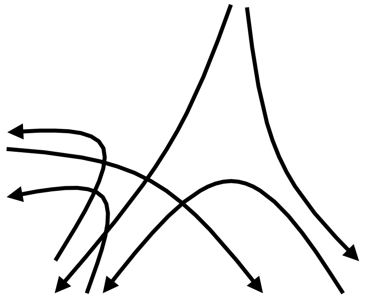


Today, providers of commercial content services predominantly use client/server based solutions

- Service quality is usually very high and predictable (availability, download time, etc.)
- But at the cost of high infrastructure expenses

Adopting P2P technology provides some opportunities

- Self-scalability during unexpected traffic peaks
- Reduction of infrastructure cost (CAPEX and OPEX)



Challenge in a commercial settings:

- *Systematically* take advantage of P2P technology and at the same time provide service guarantees

Agenda

1. Introduction
2. Single-Swarm System
    1. Deterministic Fluid Model
    2. Required Server Bandwidth
    3. Experimental Results
3. Multi-Swarm System
    1. Stochastic Model
    2. Required Aggregate Server Bandwidth
    3. Experimental Results
4. Summary

1

Introduction

Introduction

Reactive vs. Proactive Allocation Mechanisms

Two fundamental (complementary) technical approaches for content service providers using peer-assisted systems to serve customers in a cost-efficient but reliable manner:

1. Reactive Mechanisms

- Example: Periodically sample and estimate bandwidth deficit, and allocate server slots (number of connections) to meet demand
- Pitfall: Performance degradation if server is insufficiently dimensioned

2. Proactive Mechanisms

- Use system and workload models for infrastructure dimensioning and bandwidth allocation to meet expected demand
- Drawback: Requires and relies on models for request patterns and content popularities

Introduction
Workload Assumptions

Studies of Video-on-Demand, video rental (NetFlix database) and Internet video indicate:

- Power-law distribution of video popularities,
- Exponential evolution of popularity over time,
- Popularities influenced by advertisement and "hyping"

For "walled-garden" services, coarse-grained classification can help reduce complexity of model

➲ Service providers have means to model content popularity and utilize a proactive allocation mechanisms

Exploiting a priori knowledge about system and workload parameters
we have developed a tool for the
performance analysis and **server dimensioning.**

2

Single-Swarm System

A System Model

Capturing the System Capacity

- Available capacity of a hybrid P2P network:

$$C_{tot} = C_S + \eta \sum_{i \in L} c_i + \sum_{i \in S} c_i$$

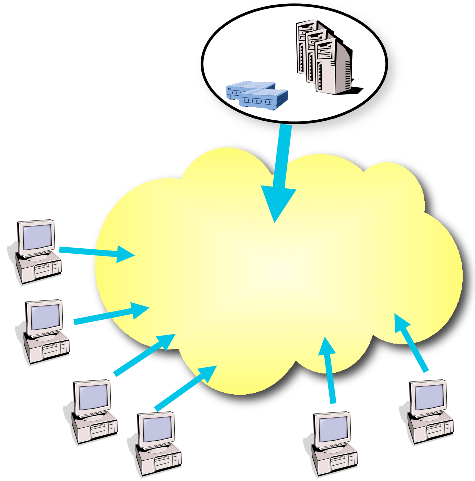
Effectiveness of leecher utilization

Uplink of peer *i*

Server capacity

Set of leecher

Set of seeders



- Time-dependent and statistical function:

$$C_{tot}(t) = C_S(t) + c(\eta x(t) + y(t))$$

- Service rate:

$$\mu(t) = \min\left\{ \frac{1}{l}\left[\frac{C_S(t)}{x(t)} + c\left(\frac{y(t)}{x(t)} + \eta \right) \right], \frac{d}{l} \right\}$$

x - number of leechers

y - number pf seeders

c - mean peer upload capacity

d - mean peer download capacity

l - file size

η - effectiveness factor

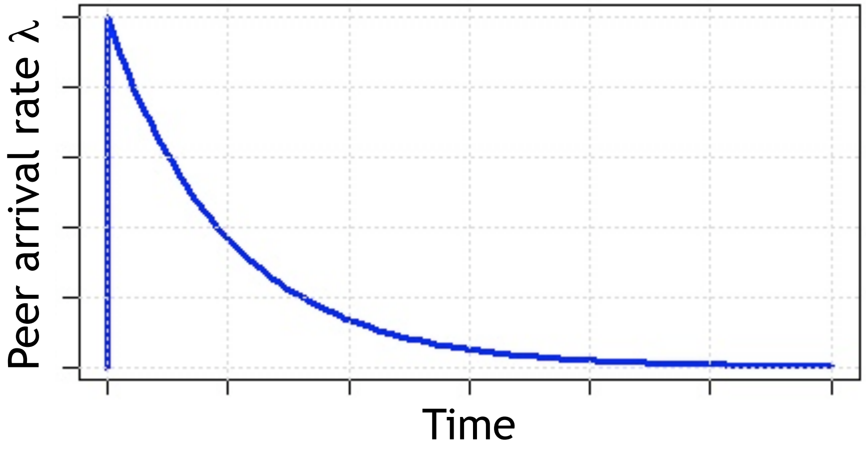
A Workload Model
Time-Decaying Arrival Process

Measurement-based studies indicate for the request arrival process:

- Does not follow Poisson
- But exhibits flash-crowd characteristics
  - Peaks Initially
  - Decays (exponentially) with time

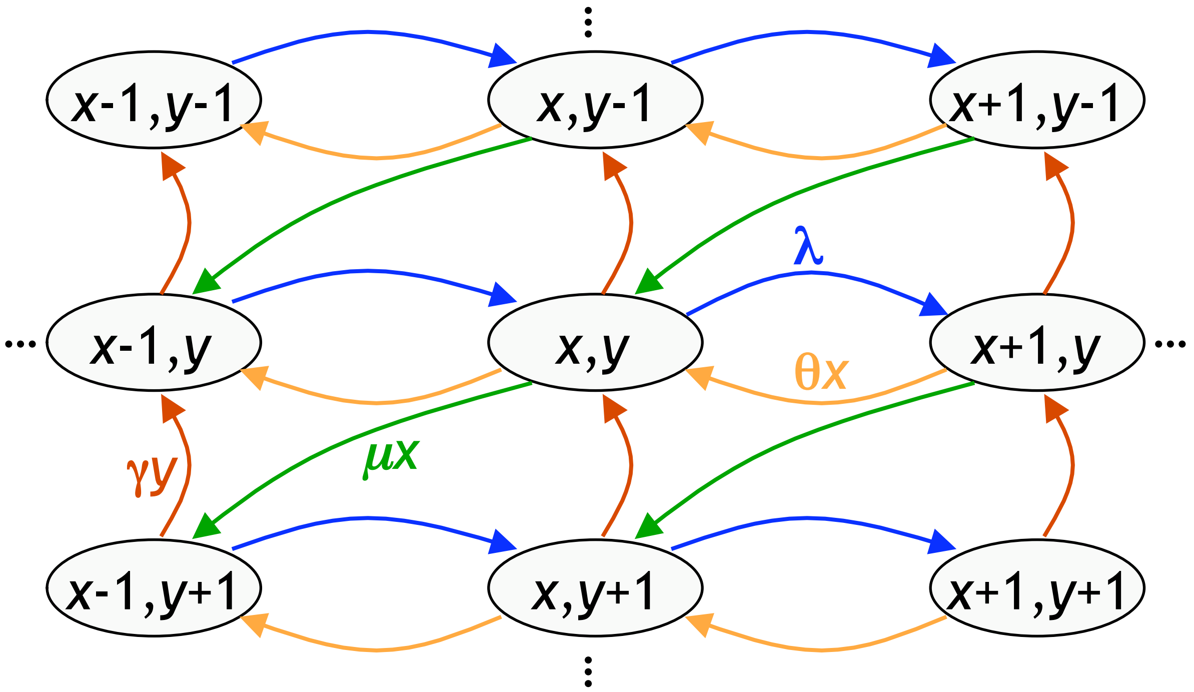
We use a request arrival model as proposed in [1]:

$$\lambda(t) = \lambda_0 e^{-\frac{t}{\tau}}$$



[1] Buo et al. “Measurements, analysis, and modeling of BitTorrent-like systems,” in Proc. of IMC 2005.

Deterministic Fluid Model



x – number of leechers

y – number of seeders

λ – peer arrival rate

θ – abandonment rate

μ – service rate

γ – seeder churn rate

Evolution of number of seeders and leechers can be expressed by LDEs:

$$\frac{dx}{dt} = \lambda - (\mu + \theta)\, x(t)$$

$$\frac{dy}{dt} = \mu x(t) - \gamma y(t)$$

$$\lambda(t) = \lambda_0 e^{-\frac{t}{\tau}}$$

$$\mu(t) = \min\left\{ \frac{1}{l}\left[\frac{C_S(t)}{x(t)} + c\left(\frac{y(t)}{x(t)} + \eta \right) \right], \frac{d}{l} \right\}$$

Required Server Capacity

Objective: Dimension server for support of a target service rate

$$\forall t \geq 0, \quad \mu \leq d/l \qquad \Rightarrow \qquad C_S^{min}(t,\mu) + c\left[\eta x(t) + y(t)\right] = l\mu x(t)$$

Deriving required server capacity using the fluid model:

$$\boxed{C_S^{\min}(t,\mu) := K(\mu)\cdot\max\left\{A_1(\mu)e^{-\frac{t}{\tau}} + A_2(\mu)e^{-(\mu+\theta)t} + A_3(\mu)e^{-\gamma t}, 0\right\}}$$

$$K(\mu) := \frac{\lambda_0}{\mu + \theta - \frac{1}{\tau}}$$

$$A_1(\mu) := \mu\left(l - \frac{c}{\gamma - \frac{1}{\tau}}\right) - c\eta$$

$$A_2(\mu) := c\eta - \mu\left(l + \frac{c}{\mu + \theta - \gamma}\right)$$

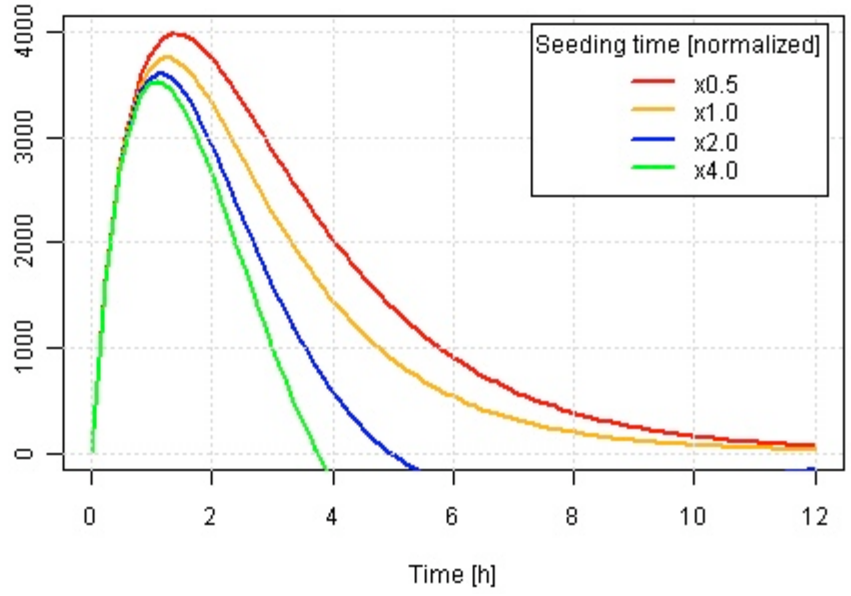
$$A_3(\mu) := c\mu\left(\frac{1}{\gamma - \frac{1}{\tau}} + \frac{1}{\mu + \theta - \gamma}\right)$$

Experimental Results

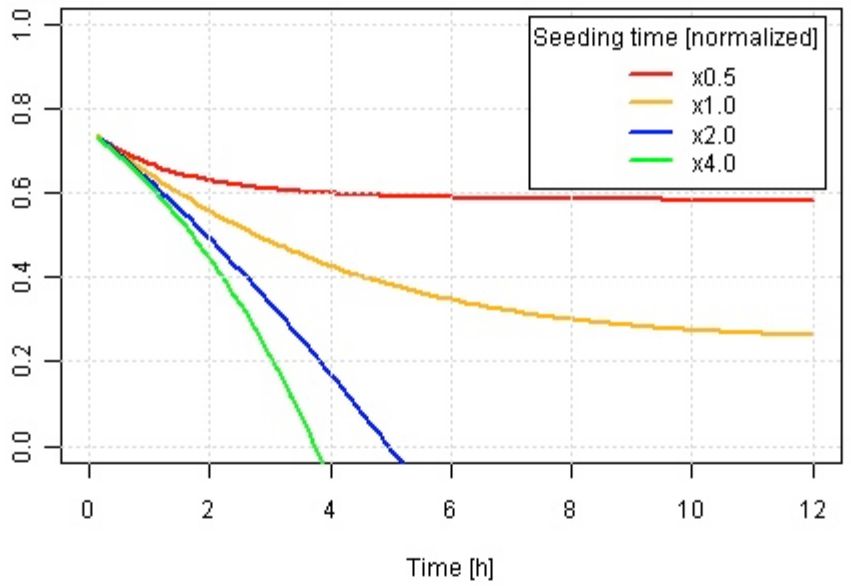
Parameter settings:

- File size l = 1.5GB
- Downlink/uplink d/c = 3.072Mbps/768kbps
- Arrival pattern: $\lambda_0=1s^{-1}$, $\tau = 2*l/d$
- Target service rate $\mu = d/l$ = 1/4000s
- Abandonment rate $\theta \rightarrow 0$
- Default seeding time $\gamma^{-1} = l/d$

Server requirement [absolute]



Server requirement normalized by requirement of a pure C/S system



3

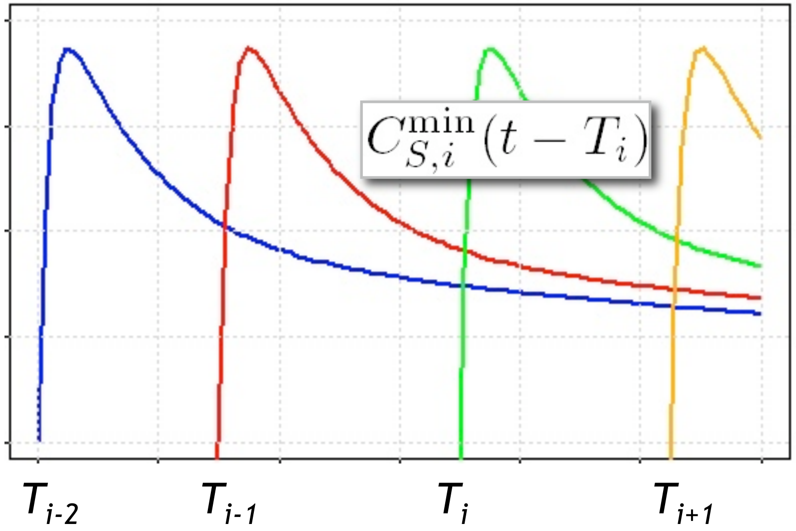
Multi-Swarm System

Multi-Swarm System
Stochastic Multi-File Model

Previously: Deterministic model for a single swarm

Next Step: Extend to multi-swarm system

- New content ingested as Poisson process of rate ν
- Server capacity for each file determined by deterministic analysis



T_i – Ingestion time of content *i*

Multi-Swarm System

Stochastic Multi-File Model (contd.)

- Total server capacity requirement at time *t*:

$$D(t) = \sum_{n=1}^{N(t)} C_{S,i}^{min}(t - T_i)$$

$N(t)$ - number of files ingested by time t

In order to achieve an adequate level of service:

- Total capacity requirement may exceed available server capacity only with a small probability

$$P\{D(t) > C\} \leq \epsilon$$

Approximation may be obtained using several methods

- Kaufman-Roberts like recursion
- Central Limit Theorem
- Chernoff bound

Experimental Setup

Parameter settings:

- File size l = 125MB
- Downlink/uplink d/c = 1.5Mbps/100kbps
- Arrival pattern: λ_0=10s^{-1}, τ = 3600s
- Target service rate μ = d/l = 1.5/1000s
- Abandonment rate θ = 0.0001
- Effectiveness η = 0.5
- Default seeding time $\gamma^{-1} = l/d$

Variable: Server capacity C

- System load maintained at ≈80%
  - Achieved by varying content ingestion rate ν

Experimental Results (contd.)

Server Capacity [Mbps]	Approx. 1 Recursion [Prob.]	Approx. 2 CLT [Prob.]	Bound [Prob.]	Sim Results [Prob.]
100	0.096150	0.0975500	0.14470	0.1266
500	0.002807	0.0018840	0.00312	0.0019
1,000	0.000047	0.0000205	0.00005	0.0000

General observation:

- Very good agreement between approximations and simulations

Interpretation of results:

- If server capacity C is scaled up & we keep load constant (increasing ν)
  ➲ $P\{D(t)>C\}$ drops
- If server capacity C is scaled and we choose to keep $P\{D(t)>C\}$ = const
  ➲ we can increase content ingestion rate ν

4

Summary

Summary

- We have developed mathematical models for hybrid P2P content distribution systems.
- We used the models to explore and demonstrate the substantial savings potential of a hybrid P2P systems.
- We gave procedures to calculate the server requirement for dissemination of content at given service rates.
- Alternatively, the models can be used to determine the maximum number of supported swarms in a given system (admission control).

Ongoing work:

- We use our models to develop resource scheduling strategies and algorithms

Thank You!