Client and Server Games in peer-to-peer networks

Iordanis Koutsopoulos

University of Thessaly, Greece

Joint work with L. Tassiulas, L. Gatzikis

YEQT, Eindhoven, Nov. 19, 2009

Department of Computer and Communications Engineering, University of Thessaly

- Belongs to school of Engineering of University of Thessaly
- Located in Volos, Greece (300km north from Athens)
- Founded in March 2000, first graduates in 2005
- 20 tenure and tenure-track faculty members, 10 visitor instructors
  - > 500 undergrad, 40 grad students

U of Thessaly team

- People
  - Leandros Tassiulas (Prof)
  - Iordanis Koutsopoulos (Assistant Prof)
  - Thanasis Korakis (Lecturer)
  - 4 post-doc researchers
  - 15 graduate students

Research

- OPNEX: FP7 STREP FIRE
  - Optimization-driven Multi-hop Network Design and Optimization
  - Start from first principles optimization theory
  - Develop decentralized algorithms spanning layers from PHY to Transport
  - Translate algorithms into implementable low-overhead protocols
- N-CRAVE: FP7-ICT-2007-1, The Network of the Future, STREP
  - Network Coding for Robust Architectures in Volatile Environments
  - Novel protocols based on Network Coding from access to application layer
  - Implementation of Network coding algorithms
- NADA: FP7 STREP FIRE
  - Nano-data centers
  - Develop a new peer-to-peer network communication paradigm based on set-top box interaction in a peer-to-peer network fashion

Research

- NEWCOM++: FP7-ICT-2007-1, The Network of the Future, NoE
  - WP10: Network Theory
  - WP8: Scheduling and adaptive RRM
  - WP11: Opportunistic wireless Networks
- NET-REFOUND: FP6-Call 5, FET OPEN (under CERTH)
  - Network Research Foundations
  - Fundamental Performance Limits of Networks
  - Ways to approximate and achieve performance limits (Network Coding, techniques migrating from Physics other sciences)
  - Network autonomy
- ONELAB: FP7 IP FIRE
  - Develop a federation of experimentation test-beds
  - Remote capability of experiments
  - Inter test-bed coordination framework

Peer-to-peer networks

- Capture dual client-server role of peers
  - Client: generates requests for content objects, which need to be satisfied by others
  - Server: uploads content to other per their request
- Client: may choose to address parts of its requests to different servers
- Server: may serve requesters with different scheduling disciplines
- Goal: Understand spontaneous interactions of rational peers wishing to exchange content
  - Selfish client peer interaction
  - Selfish client-server peer interaction
- Characterize stable network operating points emerging from peer selfish behavior (in terms of request load splits and service disciplines)

Background

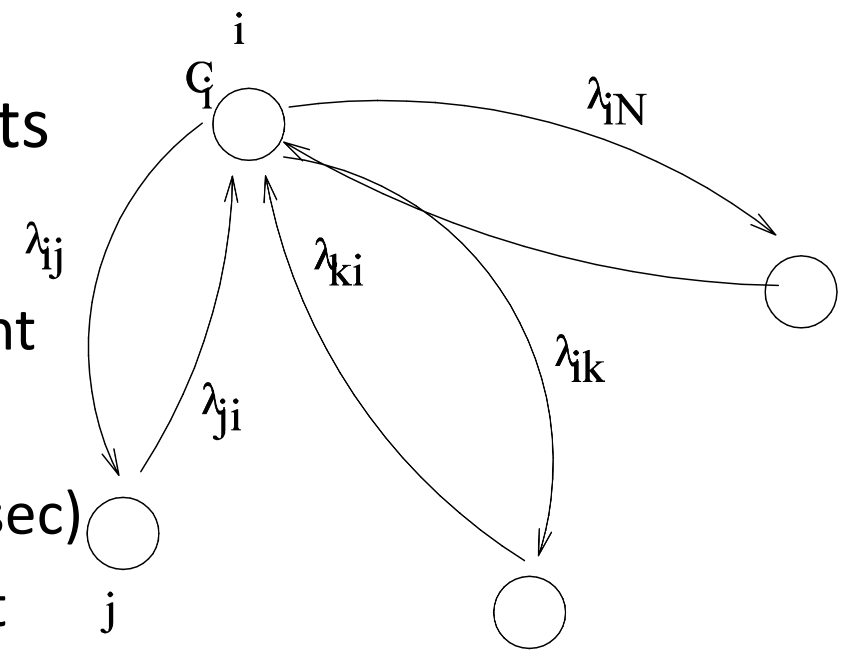
- Orda et.al [93]: Competitive routing in multiuser communication networks
  - Selfish routing of atomic user flows over parallel shared links
  - Uniqueness of NEP for certain classes of link latency functions, among which the one corresponding to M/M/1 queue with FIFO
  - Convergence to NEP shown formally for 2 users

Client and server games in peer-to-peer networks

- N peers, content exchange

– Fluid model for content requests

- Client role:
  - Peer i as client: generates content request load
  - Poisson, mean rate r_i (requests/sec)
  - Exponentially distributed request size, same mean L=1



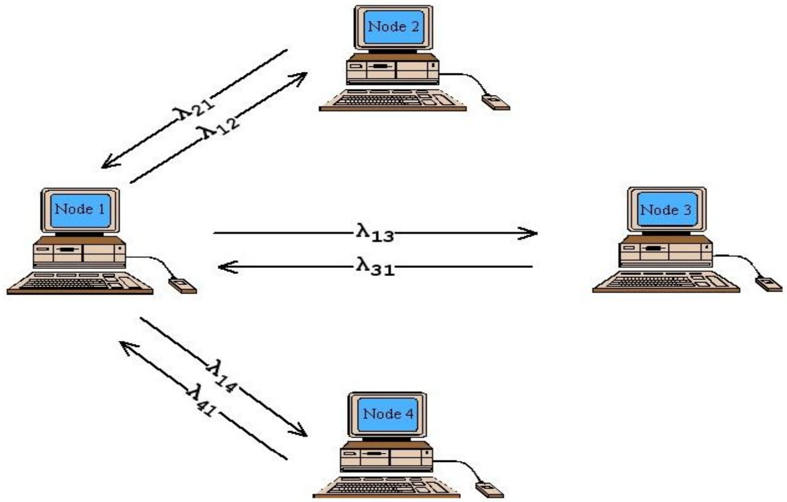
- Server role:
  - Peer i as server: Service capacity C_i (bits/sec)
  - Server is M/M/1 queue , average request service time $1/C_i$

Strategies

- *Client i strategy set, $\boldsymbol{\lambda}_i$:* set of feasible request load splits to servers,

  $r_i = \sum_{j \neq i} \lambda_{ij}$, for i=1,…N

- *Server i strategy set, $\boldsymbol{\pi}_i$:* scheduling disciplines of requests
  - FIFO: requests at server served in order of arrival
  - Preemptive priority: service based on some priority ordering. E.g. N=3, peer 1 serves with $\Pi_1 \in \{(2,3), (3,2)\}$



Model (1)

- General case
  - peer i interested in content that is available at subset of peers S_i
  - Peer i possesses content of interest to subset of peers CL_i
  - S_i (t) , CL_i (t) evolve with time
- Here, not discriminate in terms of content: S_i , CL_i ={1,…N}
- Collection of client strategies: $\boldsymbol{\Lambda} = (\boldsymbol{\lambda}_1,\ldots,\boldsymbol{\lambda}_N)$
- Collection of server strategies: $\boldsymbol{\pi} = (\boldsymbol{\pi}_1,\ldots,\boldsymbol{\pi}_N)$
- Average retrieval delay of peer i: $D_i(\Lambda,\pi) = \sum_{j\neq i} \frac{\lambda_{ij}}{r_i} D_{ij}(\Lambda,\pi_j)$
- Total average retrieval delay: $D_{\text{tot}} = \frac{1}{\sum_i r_i}\sum_{i=1}^{N} r_i D_i = \frac{1}{\sum_i r_i}\sum_{i=1}^{N}\sum_{j\neq i}^{N} \lambda_{ij} D_{ij}.$

Model (2)

- In a peer-to-peer network, retrieval delay of a client depends on:
  - Service delay at server (queuing of requests to server)
  - Overlay network topology specifics (route from server to client, congestion at intermediate nodes, …): not addressed here
- Peers in Star topology
- Average retrieval delay of client i: $D_i = \Sigma_{j \neq i} p_{ij} D_{ij}$ depends on:
  - portions of requests addressed to each server j, $p_{ij} = \lambda_{ij} / \lambda_i$
  - D_{ij} depends on service discipline at servers $j \neq i$, and loads Λ_j at servers j (load splits p_{kj} of peers $k \neq i$ to server j)
  - Consider server models:

FIFO M/M/1 queue

$$D_{ij} = \frac{1}{C_j - \Lambda_j}.$$

M/M/1 pre-emptive priority queue

$$D_{ij} = \begin{cases} \dfrac{1}{C_j - \lambda_{ij}}, & \text{if } \pi_i^j = 1, \\ \dfrac{C_j}{(C_j - \Lambda_j^{i+})(C_j - \Lambda_j^{i+} - \lambda_{ij})}, & \text{if } \pi_i^j > 1. \end{cases}$$

Research Goals

- Resource: Service capacities C_i of servers i=1,…,N
- Peers (as clients) compete in getting attention of servers
  - A peer load splitting strategy changes loads at different servers,
  - Intuitively, a peer chooses to send requests to servers j from which it experiences small delay (high service capacity C_j and light load Λ_j)
  - But then, it affects server loads and thus other peers' delays

Client Games

- Best response dynamics: peer i picks a strategy $\boldsymbol{\lambda}_i$ that optimizes some performance criterion, given others' strategies
- Parametrize peer behavioral profile based on performance criterion $$F_i^{\beta_i}(\Lambda, \pi) = \frac{r_i}{\sum_k r_k} D_i(\Lambda, \pi) + \beta_i \sum_{j \neq i} \frac{r_j}{\sum_k r_k} D_j(\Lambda, \pi).$$
  - $0 \leq \beta_i \leq 1$
  - $\beta_i = 0$ (peer is egotistic) $\beta_i = 1$ (peer is altruistic)
- Egotistic: care only for reducing own delay
- Altruistic: consider also for others' delays
- A point $\Lambda^* = (\boldsymbol{\lambda}_1^*, \ldots \boldsymbol{\lambda}_N^*)$ is NEP if for each i,

$$F_i^{\beta_i}(\lambda_i^*, \lambda_{-i}^*, \pi) \leq F_i^{\beta_i}(\lambda_i, \lambda_{-i}^*, \pi)$$

Case of Egotistic Peers (1)

- **π** = FIFO
- Each peer at each step solves: $\min_{\lambda_i \in \mathcal{F}_i} D_i(\lambda_i, \lambda_{-i}, \pi).$
  - Water-filling style Solution

$$\lambda_{ij} = C_{ij} - \frac{\sqrt{C_{ij}}}{\sum_{j\neq i}^{K_i} \sqrt{C_{ij}}} (\sum_{j\neq i}^{K_i} C_{ij} - r_i),$$

if $j \leq K_i$, and $\lambda_{ij} = 0$ otherwise, where

$$K_i = \max\{\ell : \sqrt{C_{i\ell}} \geq \frac{(\sum_{j\neq i}^{\ell} C_{ij} - r_i)}{\sum_{j\neq i}^{\ell} \sqrt{C_{ij}}}\}.$$

Residual Capacity

$$C_{ij} = C_j - \Lambda_j^{-i}$$

  - Each peer autonomously deduces C_{ij} from delay measurements
    - Measure λ_{ij} at step (n-1)
    - At each step n, measure delay per unit flow, $D_{ij}^{(n)}$
    - Compute residual capacity $C_{ij} = \lambda_{ij} + (1/D_{ij})$.
    - Compute new best response at step n
- NEP is unique [extending proof of Orda et.al., '91]
  - Sequence of best response updates converges to NEP (numerical verification)

Case of Egotistic Peers (2)

- **π** = any collective priority order profile
  - Each peer i solves again a water-filling style problem
  - Needs to know capacities C_j of servers j
    - Compute λ_{ij} at step (n-1)
    - At each step n, measure delay per unit flow, $D_{ij}^{(n)}$
    - Compute residual capacity $C_{ij}^{+} = C_j - \Lambda_j^{i+}$ as root of equation $D_{ij}^{(n)} x^2 - D_{ij}^{(n)} \lambda_{ij}^{(n)} x - C_j = 0.$
    - Rank servers according to $C_{ij}^{+} / \sqrt{C_j}$
    - Compute new best response at step n

  - NEP uniqueness
    - convergence of BR to NEP

Case of Altruistic Peers (1)

- Best response: each peer solves

$$\min_{\boldsymbol{\lambda}_i \in \mathcal{F}_i} \frac{1}{\sum_k r_k} \sum_{j \neq i} \left(\lambda_{ij} D_{ij} + \sum_{k \neq i,j} \lambda_{kj} D_{kj} \right) = \min_{\boldsymbol{\lambda}_i \in \mathcal{F}_i} D_{\mathbf{tot}}(\boldsymbol{\Lambda}, \boldsymbol{\pi})$$

- FIFO
  - Global Problem (e..g : for $C_i = C$)
  - Best response updates:
    - Each client needs to know capacities C_j of servers j
    - Measure λ_{ij} at step (n-1)
    - At each step n, measure delay per unit flow, $D_{ij}^{(n)}$
    - Compute residual capacity $C_{ij} = \lambda_{ij} + (1/D_{ij})$.
    - Compute new best response at step n

Case of Altruistic Peers (2)

- FIFO: Global Problem (e..g : for $C_i = C$)

  – Problem $\min_{\Lambda \in \mathcal{F}} D_{\text{tot}}(\Lambda) = \frac{1}{\sum_k r_k} \sum_{j=1}^{N} \frac{\Lambda_j}{C - \Lambda_j}$. falls within the class of problems

  $$\text{Problem (P)}: \quad \min_{\Lambda \in \mathcal{F}} D(\Lambda) = \sum_{j=1}^{N} G(\Lambda_j)$$

  – G(.) non-decreasing, convex in load Λ_j

  – *Any limit point of the sequence of best response updates is a NEP and also an optimal solution of the global problem*

  – Multiple NEPs, each of them inducing the same server load vector (Λ_1,…Λ_N) and same D_{tot}

- Proof: along the lines of [Hajek90] “Performance of global load balancing by local adjustment”

Case of Altruistic Peers (3)

- FIFO, Unequal capacities
  - *Any limit point of the sequence of best response updates is a NEP and also an optimal solution of the global problem*
  - D_{tot} jointly convex w.r.t. strategies
  - Proof: Best response updates are essentially Gauss-Seidel iterations

- Priority scheduling: limit points of best response are NEPs, but local minima of D_{tot}
  - D_{tot} not jointly convex w.r.t. strategies

Client - Server Games (1)

- Besides load splits, each peer i can change scheduling strategies $\boldsymbol{\pi}_i$
  - Different ensemble of scheduling strategies $\boldsymbol{\pi} = (\boldsymbol{\pi}_1,\dots,\boldsymbol{\pi}_N)$ → different best response load splits
- Peer i: find a service discipline $\boldsymbol{\pi}_i$ so that delay D_i at NEP *after* peers play a client load splitting game is minimum (service disciplines $\boldsymbol{\pi}_j$, $j \neq i$ remain fixed)
  - Idea: peer i realizes that its high delay at some server j, D_{ij}, is due to load of peer k, served with higher priority than i at server j
    - Relies on existence of voluntary signalling
  - Attract k's traffic by offering high service priority to it
    - In general: assign optimal time fraction to each priority order

Client - Server Games (2)

- Here: Two-stage client server games
  - 1st stage: a peer chooses a *"best" scheduling strategy* (in terms of optimal time fractions for different priorities)
    - Exponential number of priorities
  - 2nd stage: Peers run a client best response load splitting game and reach the (unique) NEP, and so on
  - Existence of NEP (fractions of different priorities equivalent to mixed strategies)
- Issue: need significant amount of overhead information to perform updates
- Currently investigating heuristic "good" response updates
  - Rank peers according to some metrics, related to delay reduction peers cause

Future Work

- First step: full mesh network
  - Assessed impact of selfish load splits and service disciplines on performance
  - Small Price of Anarchy PoA (3-4%) for client games
  - PoA can get larger for client-server games
  - Expect larger PoA for sparser connectivity: what is the exact relationship?
- Impact of different file sizes
- Reciprocity and agreements
- Two-stage games need to be formalized
- Mix of peer behavioral profiles (egotistic-altruistic)

- THANK YOU !

- More information:
- I. Koutsopoulos, L. Tassiulas, L. Gatzikis, Client and server games in peer-to-peer networks, IWQoS 2009.
  - telecom.inf.uth.gr
  - www.inf.uth.gr/~jordan