Computer Networks 66 (2014) 46–51

Contents lists available at ScienceDirect

Computer Networks

journal homepage: www.elsevier.com/locate/comnet

A balance of power principle for decentralized resource sharing

Yechiam Yemini

Prof. Emeritus of Computer Science, Columbia University, NYC, NY, United States

ARTICLE INFO

Keywords:
Networks
Power
Flow control
Microeconomics
Pareto-optimal
Decentralized control
Clouds

ABSTRACT

In a seminal 1978 paper Kleinrock (1979), Professor Leonard Kleinrock ("Kleinrock") derived "rules of thumb" for flow-control, using a "power" metric of delay-throughput tradeoffs. This work stimulated a stream of, still on-going (e.g., Canzian et al., 2013), flow-control researches. A particularly challenging question, still widely open, has been: how can networks optimize a global performance index, such as power and through decentralized actions?

At the same time, my Ph.D. thesis, under Len's guidance, introduced microeconomic techniques for decentralized control of channel access schemes (Yemini and Kleinrock, 1979). This research too, continues to attract current interest (e.g., Chang et al., 2013).

This paper introduces a generalized "power" metric for decentralized resource allocation, and uses it to derive novel Balance-of-Power principles, for Pareto optimal allocations. These principles substantially generalize and unify the results of Kleinrock (1979) and Yemini and Kleinrock (1979), as well as successor works, while shedding new light on resource sharing mechanisms in current virtual and cloud systems.

1. Introduction

In a seminal paper [1] Kleinrock introduced the use of "power" metric to derive deterministic rules-of-thumb, for network flow control.

It is useful to consider the rationale for this work: *"Queuing theory is hard. Most interesting queuing models cannot be solved exactly and this leaves the systems analyst in a difficult position, especially at the design phase…It is the purpose of this paper to provide some engineering rules of thumb … that use deterministic reasoning which is quite accurate and which leads to deterministic conclusions that systems can be driven close to 100% of their capacity and still perform well."*

In the past 36 years that passed since the publication of [1], queuing theory has not become simpler, but networked systems have become more complex and difficult to design, analyze and control.

This paper considers decentralized agents, competing over network resource allocations, to serve their individual workloads demands. We follow the rationale above, to derive new deterministic rules-of-thumb, for decentralized Pareto-optimal allocations. We first generalize "power", to serve as an agent's utility metric of its resource allocations. We then derive a new Balance-of-Power rule-of-thumb, to optimize the tradeoffs between selfish gains of an agent, through increasing its resource utilization, and the interference-losses, such increased utilization inflicts on other workloads.

A Pareto optimal resource allocation, balances the "lost opportunity costs", when an agent forgoes selfish increase of its resource demand, against gains of others, from reduced "interference loss".

A balance of power principle

E-mail address: yemini@cs.columbia.edu

http://dx.doi.org/10.1016/j.comnet.2014.03.011
1389-1286/

This principle, summarized above, is proved in Section 5, where both ''lost opportunity costs'' and ''interference loss'' are formally quantified.

2. Background

This section briefly introduces the ''power'' metric. The ''power'' of a service system, denoted U, [1–4,7] is defined as:

$$U(\lambda) = \lambda / T(\lambda) \quad (1)$$

where λ is the average workload arrival rate (throughput) and $T(\lambda)$ is the average time it spends in the system (delay). Power is maximized when

$$T(\lambda^*)/\lambda^* = T'(\lambda^*) \quad (2)$$

This occurs at the ''knee'' in the delay-throughput curve $T(\lambda)$, where the ray from the origin $T(\lambda)/\lambda$ is tangent to $T(\lambda)$.

For an $M/M/1$ service $U(\lambda) = \lambda/T(\lambda) = \lambda(\mu - \lambda) = \mu^2\rho(1-\rho)$, where μ is the service rate and $\rho = \lambda/\mu$ is the service utilization. Assume that time is normalized to measure in service-time units so that $\mu = 1$, then power is:

$$U(\rho) = \rho(1-\rho) \quad (3)$$

The term ρ represents utilization of the service capacity, while $(1-\rho)$ represents the *residual capacity* kept available (unused on average) to absorb random workload fluctuations. Power may be thus interpreted as a metric of trade-offs, between maximizing service utilization, while minimizing the risks of temporal congestion, due to statistical workload fluctuation.

We note several features of power maximization:

I. Power is optimized when $\rho^* = 1 - \rho^* = 0.5$; both, utilization and residual capacity use 50% of the service capacity.
II. At peak-power, the average number of customers in the system is: $N^* = \lambda^* T(\lambda^*) = \rho^*/(1-\rho^*) = 1$.
III. Furthermore, this average customer in the system spends, on average, $T(\rho^*) = 1/(1-\rho^*) = 2$ service-time units in the system; one in service and one waiting for the service to clear.

Power maximization thus results in efficient use of the service through deterministic rules: The system allocates 50% of its capacity to handle the deterministic workload average, and leaves 50% residual capacity to absorb its random fluctuations. The system is occupied by a single customer at a time, and keeps it in the queue just enough time to clear the service.

Such intuitively appealing deterministic rules-of-thumb can provide invaluable guidelines to engineer network mechanisms. It is not an accident that TCP congestion control mechanisms seek to discover the residual capacity (*sstresh*) of bottleneck nodes, through the slow-start phase; set the congestion-window to utilize 50% of this capacity; and follow with congestion-avoidance phase, to adapt to changes in the bottleneck residual capacity.

Subsequent work (e.g., [3,9–11,**?**,17]) expanded and applied these results broadly. However, optimizing power for an entire network, involving multiple interfering flows, turned out to be an elusive challenge (e.g., [9]). A global power metric may not admit decentralized optimization by local nodes. Nodes may be unable to monitor and control interference between flows at remote nodes, based on local information and actions.

These limitations of optimizing global power metrics suggest using decentralized optimization techniques. Such techniques, developed by microeconomics [25], consider resource-sharing agents, competing to maximize their individual utilities. In the context of network flows, flows compete over shared link allocations; the power of a flow defines its utility. Power maximization becomes a multi-objective optimization problem, resolved by Pareto-optimization techniques described below.

Pareto optimization was first applied to network resource sharing in [5,8], then expanded to a broad variety of network resource allocations [13–16,18–20,23]. Starting in the mid 1990s the economics of network has attracted significant interest and applied in thousands of researches, and even several products (e.g., [26]).

3. Pareto optimal resource allocation

Consider a resource allocated among n agents, with the k-th agent getting a fraction $0 \leqslant X_k \leqslant 1$ of the resource capacity. A vector allocation $\mathbf{X} = (X_1, X_2, \ldots, X_n)$ creates utility value to the k-th agent: $0 \leqslant U_k(\mathbf{X})$. Agents compete over resource allocations, to optimize their utilities.

The utility mapping $\mathbf{U}(\mathbf{X})$ transforms resource allocations in utilization space $X = [0,1]^n$ to utilities space $U = [0,\infty)^n$. A utility vector $\mathbf{U}$ dominates $\mathbf{Y}$, denoted $\mathbf{Y} < \mathbf{U}$, if $Y_k \leqslant U_k$ for every agent k, with at least one strict inequality. A utility vector that is not dominated by any other utility vector is called *Pareto optimal* (or, ''efficient''). An allocation $\mathbf{X}$, yielding a Pareto optimal utility $\mathbf{U}(\mathbf{X})$, is called *Pareto optimal allocation*.

Loosely speaking, a Pareto optimal allocation is one where no agents can improve their utilities, without having some other agents see their utilities decrease. Under broad assumptions on the regularity and smoothness of the mapping $\mathbf{U}(\mathbf{X})$, the set of Pareto optimal allocations forms a surface within the allocations space X, called the *Pareto frontier* (see [24,25]). Pareto optimization is concerned with characterizing and computing Pareto optimal allocations and the Pareto frontier.

We proceed to apply Pareto-optimization techniques to decentralized power optimization.

Example 1 (*Two link sharing flows*). Consider, two $M/M/1$ flows, sharing a network link. Let X_k denote the fraction of the link capacity utilized by flow-k ($k = 1, 2$). It is convenient to use alternate term to describe X_k: ''allocation'', ''resource share'', or ''workload'' of flow-k. An allocation $\mathbf{X} = (X_1, X_2)$ yields a power utility for each flow: $U_k = X_k(1 - X_1 - X_2)$.

What are the Pareto optimal allocations of $\mathbf{U}(\mathbf{X})$?

Suppose, first, that the two flows agree to share the link equally: $X_1 = X_2 = x$. This yields the power utility functions $U_k(x) = x(1-2x)$, optimized at $x = 1/4$. This fair allocation