Insensitive Traffic Splitting in Data Networks

Juha Leino[a] and Jorma Virtamo[a]

[a]Networking Laboratory, Helsinki University of Technology,
P.O. Box 3000, FIN-02015 HUT, Finland
{Juha.Leino,Jorma.Virtamo}@tkk.fi

Abstract: Bonald et al. have studied insensitivity in data networks assuming a fixed route for each flow class. If capacity allocation and routing are balanced and the capacity of a given class is shared equally between the flows, the network state distribution and flow level performance are insensitive to any detailed traffic characteristics except the traffic loads. In this paper, we consider optimal insensitive load balancing executed at packet level so that the traffic of each flow may be split over several routes. Similarly to the case with fixed routing, the most efficient capacity allocation and traffic splitting policy can be determined recursively. We formulate the problem as an LP problem using either a set of predefined routes or arbitrary routes and present numerical results for two toy networks. Traffic splitting gives a clear performance improvement when compared to flow level balancing or fixed shortest path routing.

Keywords: Insensitivity, Traffic splitting, Whittle networks

1. INTRODUCTION

Load balancing has important applications in many computer and communication systems. Performance of a system is improved, if the service demands are divided between the servers in an efficient way. In static load balancing, the balancing does not depend on the system state and the optimal policy can be determined as a simple optimization problem. Better performance is obtained if the balancing depends on the system state. Dynamic load balancing problem is more complicated as the optimal policy is sensitive to detailed customer characteristics such as job size distribution. Optimal dynamic load balancing is a difficult problem and even simple systems can have nontrivial solutions [1].

In this paper, we discuss load balancing in data networks. In general, performance evaluation of data networks is difficult because detailed traffic characteristics such as flow size distribution affect the system behaviour. Data networks can be modeled as open processor sharing networks in which customers represent data flows. Recently, Bonald and Proutière have modeled networks using Whittle networks [2,3]. If the capacity allocation is balanced and session arrivals are Poissonian, the steady state distribution is insensitive, i.e. does not depend on any traffic characteristics except the traffic loads on different routes. They introduced the concept of balanced fairness (BF) as the most efficient capacity allocation policy when static routing is used. The concept of balanced fairness has been generalized in order to analyze the performance of wireless ad hoc networks [4].

Load balancing can be executed at two different levels. Either an arriving flow is directed to a route and the same route is utilized until the flow is finished or a flow can be split between several routes. The first approach corresponds to flow level load balancing and the second to packet level load balancing. The feasibility of these methods depend on the studied network. Flow level balancing is a technically feasible solution in TCP based networks. Traffic splitting is not feasible in current TCP networks as different delays in different routes mix up the packet order. However, other approaches tolerate delay differences, for example protocols utilizing digital fountain codes do not depend on packet ordering [5].

Bonald and Proutière introduced the idea of insensitive flow level load balancing in [2]. When flow level balancing is used, an arriving flow is directed to one of the routes and the same route is utilized until the flow is finished. Optimal insensitive routing policies have been identified utilizing either local [6] or global [7] state information. Better performance is achieved if both capacity allocation and routing are optimized jointly [7,8]. In this paper, we discuss insensitive load balancing at packet level. When balancing is executed at packet level, the packets of a flow can be divided among several routes.

The best balanced capacity allocations can be determined recursively. Starting from an empty network, the amount of utilized bandwidth is maximized in every state while satisfying two sets of constraints. The network imposes capacity constraints and insensitivity requires balance condition to be satisfied. The main contribution of this paper is the formulation of the maximization problem in the case of traffic splitting. When flows can be split onto multiple routes, the bandwidth maximization can be solved as a linear programming (LP) problem using a formulation based on network flows. In some special cases discussed in section 4, there is no need to solve the actual LP problem but the maximal allocation can be found using methods based on minimum cuts. The minimum cut approach can be used to determine bounds or approximations for the network performance.

We assume that the network has an access control policy that rejects an arriving flow if a minimum usable bandwidth cannot be provided. Blocking probability is then used as a performance metric in the numerical examples presented in section 5.

2. INSENSITIVITY IN DATA NETWORKS

2.1. Network Model

We consider a network consisting of a set of nodes $\mathcal{N}$ and links $\mathcal{L}$. The capacity of link $l \in \mathcal{L}$ is C_l. We assume that there are $\mathcal{K}$ classes of flows. The flows are elastic, i.e. the size of a transfer is fixed and the duration depends on the allocated bandwidth. The traffic load (bits/s) of class-k is ρ_k. The state of the network is defined by the vector $x = (x_1, \ldots, x_{\mathcal{K}})$, where x_k is the number of class-k flows in progress.

As discussed in [3], the modeling approach allows the flows to be generated within sessions. A session is composed of a random number of flows and think times. The flow sizes and think time durations can have arbitrary distributions and may be correlated. The session arrival process of every class is assumed Poissonian. Poissonian session arrivals correspond to a large number of independent users. The validity of the arrival process is supported by Internet traffic measurements [9].

The bandwidth allocated for class-k flows is denoted $\phi_k(x)$ and depends on the network state. The bandwidth of a class is divided equally between the flows in that class. A network is subject to some capacity constraints and a feasible allocation has to satisfy these constraints. A typical example is that allocated capacities may not exceed link capacities.

2.2. Insensitivity

Assuming Poissonian session arrivals and static routing, a network is insensitive if and only if capacity allocation is balanced, i.e. it satisfies the balance condition [3]

$$\frac{\phi_i(x-e_j)}{\phi_i(x)} = \frac{\phi_j(x-e_i)}{\phi_j(x)} \quad \forall i, j, x_i > 0, x_j > 0, \tag{1}$$

where e_i is a vector with 1 in component i and 0 elsewhere. An allocation is balanced if and only if there exists a balance function $\Phi(x)$ so that $\Phi(0) = 1$ and

$$\phi_i(x) = \frac{\Phi(x-e_i)}{\Phi(x)} \quad \forall x_i > 0. \tag{2}$$

The higher the value of the factor $\Phi(x)^{-1}$ in (2), the more bandwidth is utilized in state x. Balanced allocation with highest bandwidth can be determined recursively. The bandwidth ratios of the classes in state x are fixed by the earlier values of $\Phi(x-e_i)$, $x_i > 0$. A network imposes some constraints on the maximum bandwidths. In order to determine the most efficient capacity allocation, $\Phi(x)^{-1}$ is increased until a constraint is met.

If the routes of the traffic classes are fixed, the only balanced allocation saturating at least one link in every state is balanced fairness as defined in [3]. Class-k flows utilize route r_k which is a subset of links $r_k \subset \mathcal{L}$. BF is defined as $\Phi(0) = 1$ and

$$\Phi(x) = \max_l \left\{ \frac{1}{C_l} \sum_{k:l \in r_k} \Phi(x-e_k) \right\}. \tag{3}$$

The problem of maximizing $\Phi(x)^{-1}$ is reduced to finding the link that is saturated first, i.e. that realizes the maximum of (3).

The steady state distribution of the system is

$$\pi(x) = G^{-1}\Phi(x) \prod_{k=1}^{\mathcal{K}} \rho_k^{x_k}, \tag{4}$$

where G is the normalization constant [2]. The state distribution depends on the traffic characteristics only through the traffic loads ρ_k of the classes.

3. OPTIMAL INSENSITIVE TRAFFIC SPLITTING

If the traffic can be split onto different routes, more capacity can be allocated to the flows than with fixed routes. The splitting problem can be defined in two ways. Either there is a predefined set of routes for each traffic class or the routes are arbitrary. In both cases, the maximal amount of allocated capacity is unambiguous, but the there can be several ways to provide the capacity to the traffic classes. As an example, several parallel links limit the amount of traffic, but the traffic classes can be split onto the links in different ways.

3.1. Problem with Predefined Routes

We assume that class-k flows can be split onto routes $r \in R_k$. Each route r consists of a set of links $r \subset \mathcal{L}$. The bandwidth allocated for class-k flows on route r is denoted $\phi_k^r(x)$. The total bandwidth allocated for class-k traffic is $\phi_k(x) = \sum_{r \in R_k} \phi_k^r(x)$. The allocations have to satisfy the capacity constraints

$$\sum_k \sum_{r \in R_k : l \in r} \phi_k^r(x) \leq C_l \quad \forall x, l. \tag{5}$$

The allocated capacity is maximized recursively for all the states. The problem of finding the maximal capacity allocation in a given state x while satisfying the balance condition (1) and the feasibility condition (5) can be formulated as a linear optimization problem. To simplify the notation, we define $u = \Phi(x)^{-1}$ for a given state x. The formulation is

$$\max_{u, \phi_k^r} \quad u \tag{6}$$

$$\text{s.t.} \quad \sum_{r \in R_k} \phi_k^r(x) = u\,\Phi(x - e_k) \quad \forall k : x_k > 0, \tag{7}$$

$$\sum_k \sum_{r \in R_k : l \in r} \phi_k^r(x) \leq C_l \quad \forall l, \tag{8}$$

$$\phi_k^r(x) \geq 0 \quad \forall k, r, \tag{9}$$

where (7) is the balance condition and (8) represents the capacity constraints. The problem can be solved using standard LP algorithms. If there is only one route per class the optimal allocation is identical with the ordinary balanced fairness and can be solved using recursion formula (3).

3.2. Problem with Arbitrary Routes

A more general problem can be formulated by not assuming predefined routes. Capacity is utilized more efficiently, if all possible routes can be utilized instead of a set of predefined ones. In each state, the amount of traffic is maximized over all possible routes while satisfying the capacity and balance constraints.

Similarly with the problem with predefined routes, the problem can be formulated and solved as an LP problem. The amount of class-k traffic on the link from node i to node j is denoted ϕ_k^{ij}. The link capacity between nodes i and j is C_{ij}. The problem formulation is

$$\max_{u, \phi_k^{ij}} \quad u \tag{10}$$

$$\text{s.t.} \quad \sum_j \phi_k^{ij}(x) - \sum_j \phi_k^{ji}(x) = \begin{cases} u\,\Phi(x - e_k), & i = s_k \\ 0, & \forall i \neq s_k, t_k \\ -u\,\Phi(x - e_k), & i = t_k, \end{cases} \quad \forall k \in K \tag{11}$$

$$\sum_k \phi_k^{ij}(x) + \sum_k \phi_k^{ji}(x) \leq C_{ij} \quad \forall i, j, \tag{12}$$

$$\phi_k^{ij}(x) \geq 0 \quad \forall i, j, k, \tag{13}$$

where s_k and t_k are the source and destination of class-k flows.

The number of the variables and constraints can be reduced by aggregating the traffic classes originating from common source nodes. The smaller problem results in shorter computation times. The amount of traffic originating from node s on the link from node i to node j is denoted $\phi_{(s)}^{ij}(x) = \sum_{k:s_k=s} \phi_k^{ij}(x)$. Let the set of destinations of traffic classes originating from node s be $T_s = \{n \in \mathcal{N} \mid \exists k \text{ s.t. } s_k = s \text{ and } t_k = n\}$. The problem can thus be formulated as

$$\max_{u,\phi_{(s)}^{ij}} \quad u \tag{14}$$

$$\text{s.t.} \quad \sum_j \phi_{(s)}^{ij}(x) - \sum_j \phi_{(s)}^{ji}(x) = \begin{cases} 0, & \forall i \notin \{s, T_s\} \\ -u\,\Phi(x - e_{k:s_k=s \wedge t_k=i}), & \forall i \in T_s, \end{cases} \quad \forall s \tag{15}$$

$$\sum_s \phi_{(s)}^{ij}(x) + \sum_s \phi_{(s)}^{ji}(x) \le C_{ij} \quad \forall i,j \tag{16}$$

$$\phi_{(s)}^{ij}(x) \ge 0 \quad \forall i,j,s. \tag{17}$$

The flows of the classes $\phi_k^{ij}(x)$ can easily be determined when the aggregated flows $\phi_{(s)}^{ij}(x)$ have been solved by considering a network where the capacity of a directed link from node i to node j is given by $\phi_{(s)}^{ij}(x)$. The maximal value of u is unambiguous, but there can be numerous ways to select the routes of the aggregate flows. In the same way, there can be numerous ways to divide the aggregate flows $\phi_{(s)}^{ij}(x)$ into the flows of the individual classes $\phi_k^{ij}(x)$.

4. SOLVING THE LP PROBLEM USING MINIMUM CUTS

The problem with arbitrary routes is closely related to network flow problems, see e.g. [10], and some of the knowledge in this field can be used to gain insight into our problem. In a given state x, the aim of the optimization problem is to maximize the total traffic flow from the sources to destinations while satisfying the link capacity constraints and the balance condition fixing the ratios of allocations for different classes. The problem corresponds to a network flow problem called concurrent max-flow problem [11]. Each commodity k has a demand D_k between a source node $s_k \in \mathcal{N}$ and a sink $t_k \in \mathcal{N}$. Constant u is maximized so that the fraction uD_k of each flow is transfered. In our balanced splitting problem, the demands are $D_k = \Phi(x - e_k)$ and $u = \Phi(x)^{-1}$.

The concurrent multicommodity problem can be formulated and solved as an LP problem as seen in section 3.2. However, specialized network algorithms are significantly faster than general LP solvers in many specific problem classes. Several maximum flow problems can be solved using minimum cuts. The seminal work of Ford and Fulkerson showed that the maximum flow always equals the capacity of the minimum cut separating the source from the destination in the single commodity maximum flow problem [12]. The concept of minimum cut can be generalized for multicommodity flows as

$$\rho^* = \min_{S \subset \mathcal{N}} \frac{\sum_{i,j \in \mathcal{N}:|S \cap \{i,j\}|=1} C_{ij}}{\sum_{k \in K:|S \cap \{s_k,t_k\}|=1} D_k}. \tag{18}$$

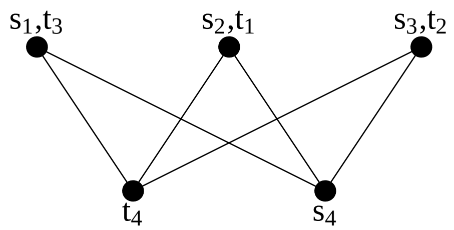


Figure 1. A graph with min-cut 1 and max-flow 3/4. All demands and capacities are one. [15]

The minimum cut equals the maximum flow u for 2-commodity flows. In general, the maximum flow can be smaller than the minimum cut as Figure 1 illustrates. With more than two commodities, the equality holds for networks with a single source and multiple sinks. In networks with undirected links, the equality holds also with a single sink and multiple sources. The maximum flow minimum cut equality has been proven for many special classes of networks, see e.g. [13–15]. If the equality holds for a given network, it is sufficient to find the minimum cut in order to determine the constant u. This is a more straightforward approach than to solve the corresponding LP problem and leads to a recursion similar to balanced fairness defined in (3). The recursion is $\Phi(0) = 1$ and

$$\Phi(x) = \max_{S \subset N} \frac{\sum_{k \in K : |S \cap \{s_k, t_k\}| = 1} \Phi(x - e_k)}{\sum_{i,j \in \mathcal{N} : |S \cap \{i,j\}| = 1} C_{ij}}. \tag{19}$$

Max-flow min-cut results can be used to derive bounds for the concurrent multicommodity problem. According to our knowledge, the tightest lower bound for constant u is [16]

$$u \geq \frac{1}{c \lceil \log k^* \rceil} \rho^*, \tag{20}$$

where c is a constant and k^* is the cardinality of the minimal vertex cover of the demand graph, i.e. the minimum number of nodes that include either the source or the sink of every source-sink pair. The lower bound can be used to determine performance bounds for insensitive traffic splitting, if the max-flow min-cut equality does not hold.

5. NUMERICAL EXAMPLES

In this section, we provide numerical results in simple toy networks. Packet level flow balancing is compared with flow level balancing and shortest path routing.

5.1. Triangle Network

First, we consider a network consisting of three nodes illustrated in Figure 2. It is fully connected with unit capacity links. Traffic loads between all node pairs are equal and we assume unit mean flow size. Total offered flow arrival intensity is denoted λ_o. Since there are only two routes between each node pair, the formulations with predefined or arbitrary

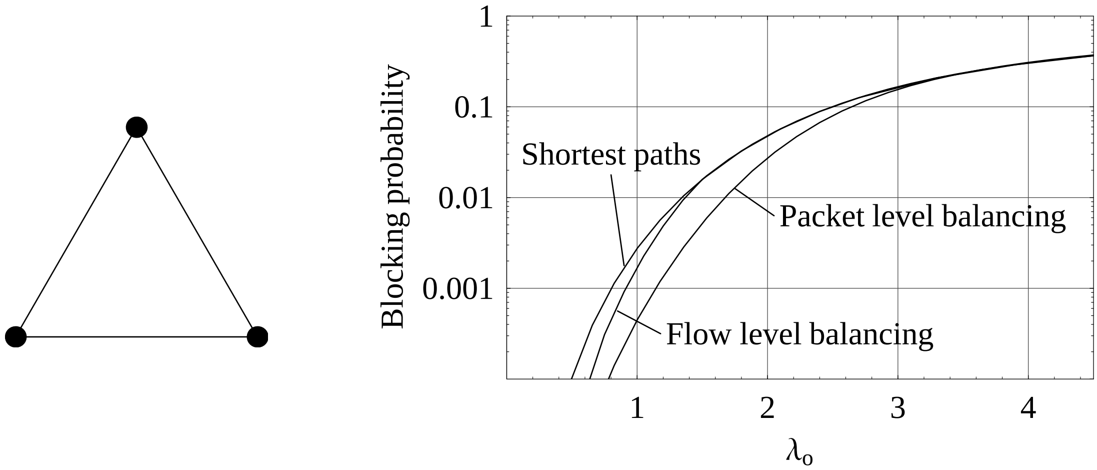


Figure 2. Complete graph with three nodes and blocking probabilities with fixed routes and load balancing.

routes do not differ. The network satisfies the criteria in [15], hence the min-cut max-flow theorem can be used.

We assume that an admission control policy rejects offered flows if a minimum bandwidth $b^{\min} = 1/5$ cannot be provided. In order to evaluate performance, we determine the overall blocking probability of flows in the system. The performance is compared with shortest path routing without load balancing and with insensitive load balancing executed at flow level. The flow level load balancing was introduced in [7]. The capacity is allocated according to balanced fairness and the flows are divided among the routes so that the system is insensitive. The best such routing policy can be determined using Markov decision theory. It should be noted, that the flow level approach assumes Poissonian flow arrivals while the traffic splitting discussed in this paper assumes only Poissonian session arrivals. The blocking probabilities with different loads are illustrated in Figure 2 demonstrating that packet level balancing performs the best. Flow level balancing outperforms the static system only with low traffic loads.

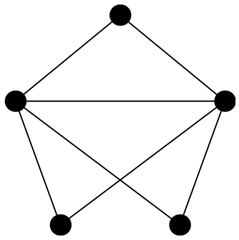
5.2. Network with Five Nodes

A more complex network with five nodes is illustrated in Figure 3. We assume that the traffic loads between all node pairs are equal and that the links have unit capacity. The minimum bandwidth is $b^{\min} = 1/3$. The blocking probabilities using shortest path routing and traffic splitting are illustrated in Figure 3.

Applying Little's formula, we get the mean transmission duration of an accepted flow

$$\mathrm{E}[T] = \frac{\mathrm{E}[|X|]}{\lambda_a} = \frac{\mathrm{E}[|X|]}{(1-B)\lambda_o}, \tag{21}$$

where $\mathrm{E}[|X|]$ is the mean number of active flows, λ_a is the accepted flow arrival intensity and B is the blocking probability. Figure 4 illustrates the mean duration of an accepted flow as a function of the blocking probability. It should be noted that a network utilizing



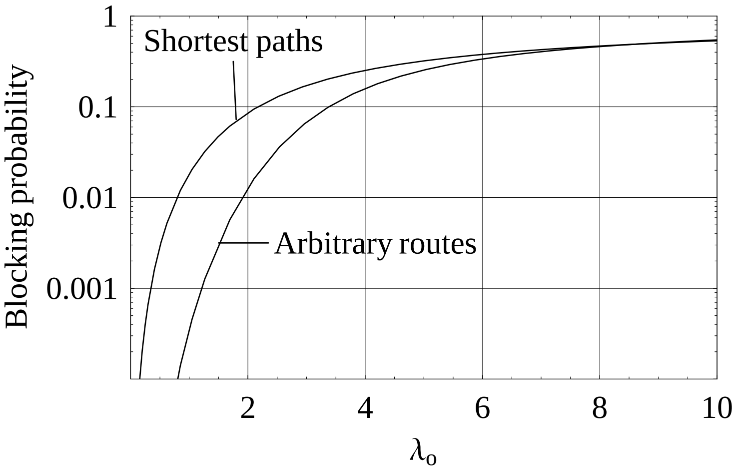


Figure 3. Example network with five nodes and blocking probabilities with fixed routes and traffic splitting.

traffic splitting carries more traffic with a given blocking probability. For comparison, Figure 5 illustrates the mean duration of an accepted flow as a function of accepted flow arrival intensity. In both cases, traffic splitting reduces the durations significantly. The advantage decreases as the amount of traffic increases.

6. CONCLUSIONS

Optimal load balancing is difficult because optimal policy and performance depend on detailed traffic characteristics. Bonald et al. have studied insensitivity in data networks assuming a fixed route for each flow class. If capacity allocation is balanced and the capacity of a given class is shared equally between the flows, state distribution and flow level performance depend only on the traffic loads. The most efficient insensitive allocation, balanced fairness, can be determined recursively.

Recently, insensitive load balancing has been considered at flow level. In this paper, we analyzed insensitive load balancing executed at packet level. Instead of routing an arriving flow into a fixed route, the traffic of the flow may be split over several routes. Similarly to the case with fixed routes, the state distribution and flow level performance is insensitive to any detailed traffic characteristics if balanced capacity allocation is used. We presented a recursive method for finding the optimal load balancing policy utilizing either a set of predefined routes or using arbitrary routing. In every state, the amount of allocated capacity is maximized by solving an LP problem. In order to reduce computation time, it was formulated using aggregated traffic flows. In some special cases, it is sufficient to solve a minimum cut problem instead of the LP problem.

We illustrated the performance of traffic splitting in two simple networks. Blocking probabilities and mean transmission durations were compared to fixed shortest path routing and to optimal insensitive flow level load balancing. Traffic splitting resulted in lower blocking probabilities and flow durations.

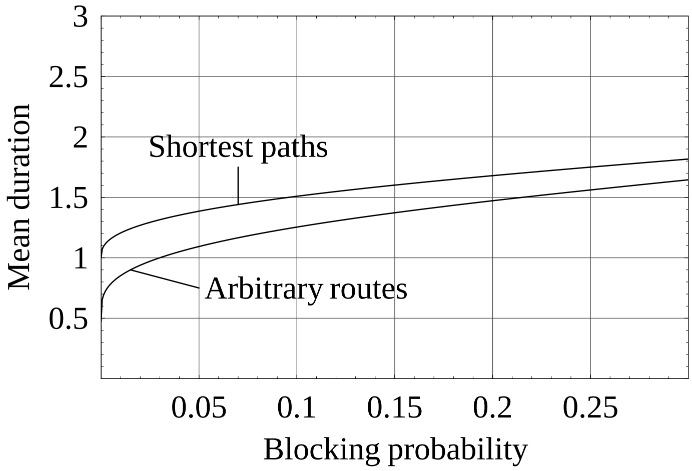


Figure 4. Mean duration of an accepted flow.

REFERENCES

1. W. Whitt. Deciding which queue to join: some counterexamples. *Operations Research*, 34:226–244, 1986.
2. T. Bonald and A. Proutière. Insensitivity in processor-sharing networks. *Performance Evaluation*, 49:193–209, 2002.
3. T. Bonald and A. Proutière. Insensitivite bandwidth sharing in data networks. *Queueing Systems and Applications*, 44:69–100, 2003.
4. A. Penttinen and J. Virtamo. Performance of wireless ad hoc networks under balanced fairness. In *Proceedings of Networking 2004*, pages 235–246, 2004.
5. J. W. Byers, M. Luby, M. Mitzenmacher, and A. Rege. A digital fountain approach to reliable distribution of bulk data. In *Proceedings of the ACM SIGCOMM '98*, pages 56–67, 1998.
6. T. Bonald, M. Jonckheere, and A. Proutière. Insensitive load balancing. In *Proceedings of Sigmetrics/Performance 2004*, pages 367–377, 2004.
7. J. Leino and J. Virtamo. Optimal load balancing in insensitive data networks. In *Proceedings of QoS-IP 2005*, pages 313–324, 2005.
8. M. Jonckheere and J. Virtamo. Optimal insensitive routing and bandwidth sharing in simple data networks. In *Proceedings of Sigmetrics 2005 (to appear)*, 2005.
9. A. Feldmann, A. C. Gilbert, W. Willinger, and T. G. Kurtz. The changing nature of network traffic: Scaling phenomena. *Computer Communication Review*, 28:5–29, 1998.
10. R. K. Ahuja, T. L. Magnanti, and J. B. Orlin. *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, 1993.
11. T. Leighton and S. Rao. Multicommodity max-flow min-cut theorems and their use in designing approximation algorithms. *Journal of the ACM*, 46:787–832, 1999.
12. L. R. Ford and D. R. Fulkerson. Sur le probléme des courbes gauches en topologie. *Canadian Journal of Mathematics*, 8, 1956.

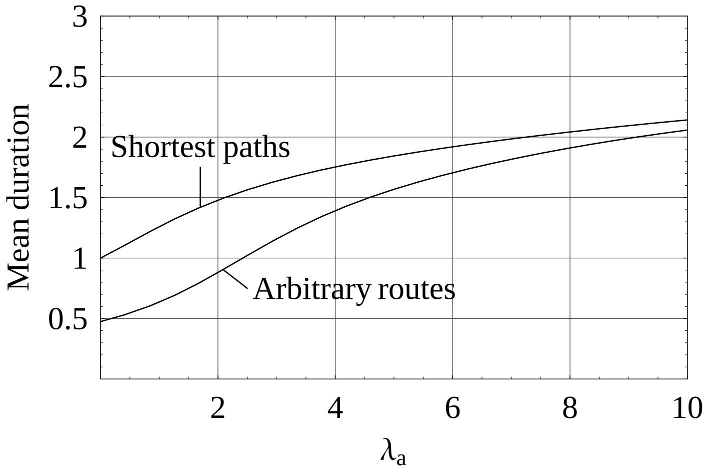


Figure 5. Mean duration of an accepted flow.

13. H. S. Zullo. *Feasible flows in multicommodity graphs.* PhD thesis, University of Colorado at Denver, 1995.
14. A. Schrijver. *Paths, Flows, and VLSI-Layout*, chapter Homotopic routing methods, pages 329–371. Springer-Verlag, Berlin, Germany, 1990.
15. H. Okamura and P. D. Seymour. Multicommodity flows in planar graphs. *Journal of Combinatorial Theory*, 31, 1981.
16. O. Günlük. A new min-cut max-flow ratio for multicommodity flows. In *Proceedings of IPCO 2002*, pages 54–66, 2002.