G Model
AEUE-50930; No. of Pages 10
ARTICLE IN PRESS
Int. J. Electron. Commun. (AEÜ) xxx (2012) xxx–xxx
Contents lists available at SciVerse ScienceDirect
International Journal of Electronics and Communications (AEÜ)
journal homepage: www.elsevier.com/locate/aeue

Bottlenecks and stability in networks with contending nodes

Tom Coenen[a,*], Hans van den Berg[b,c], Richard J. Boucherie[a], Maurits de Graaf[a,d], Ahmad Al Hanbali[e]

[a] Department of Applied Mathematics, University of Twente, Enschede, Netherlands
[b] TNO Information and Communication Technology, Delft, Netherlands
[c] Department of Design and Analysis of Communication Systems, University of Twente, Enschede, Netherlands
[d] Thales Nederland B.V., Huizen, Netherlands
[e] Department of Management and Governance, University of Twente, Enschede, Netherlands

ARTICLE INFO

Article history:
Received 24 August 2010
Accepted 14 June 2012

Keywords:
Bottleneck
Stability
Contention

ABSTRACT

This paper considers a class of queueing network models where nodes have to contend with each other to serve their customers. In each time slot, a node with a non-empty queue either serves a customer or is blocked by a node in its vicinity. The focus of our study is on analyzing the throughput and identifying bottleneck nodes in such networks. Our modeling and analysis approach consists of two steps. First, considering the slotted model on a longer timescale, the behavior is described by a continuous time Markov chain with state dependent service rates. In the second step, the state dependent service rates are replaced by their long run averages resulting in an approximate product form network. This enables us to determine the bottleneck nodes and the stability condition of the system. Numerical results show that our approximation approach provides very accurate results with respect to the maximum throughput a network can support. It also reveals a surprising effect regarding the location of bottlenecks in the network when the offered load is further increased.

1. Introduction

Inspired by wireless ad hoc networks where interference prohibits neighboring nodes to simultaneously transmit packets, this paper considers a class of open queueing network models in which servers contend for service slots. In each time slot nodes that have packets available for transmission try to obtain the channel to transmit their packets. As nodes within each others interference range cannot transmit at the same time, an allocation mechanism, i.e. a medium access control protocol, is used to decide which nodes get the opportunity to transmit, i.e. to serve a packet. Once a server in a node is allowed to transmit a packet, it blocks the servers in a specified set of other nodes corresponding to an interference neighborhood. Upon service completion, a packet either moves to a next node for further service, or leaves the network. The network is called stable when for each node the average service rate exceeds the average arrival rate of packets. When multiple or large flows pass through a node, the service rate of the node may not suffice, making this node a bottleneck. This paper investigates the stability range, the arrival rates of flows at which nodes become bottlenecks, and the throughput of the network.

The behavior of the system under consideration can be described by a state dependent discrete time Markov chain as we assume that in each slot the contention between nodes takes place independent of previous outcomes. Inspired by results obtained for loss-networks, we make a two step approximation to analyze this network, see Fig. 1. As a first step, we consider the long term average behavior, which neglects the effect of the slotted time and leads to a continuous time Markov chain. However, with the transition rates of this chain still being state dependent, analysis remains cumbersome and a further approximation is needed. Using a long term average service rate, we introduce a product form network approximation which enables us to find the bottlenecks in networks of arbitrary size and topology and determine the maximal throughput. Interestingly, it turns out that when the load of the network is increased, a bottleneck node can become stable again as a different node becomes the bottleneck. This surprising behavior is predicted correctly by our product form model.

The remainder of the paper is organized as follows. First, Section 2 gives a literature overview, after which Section 3 introduces the discrete model and contention process. Section 4 describes the first approximation step resulting in the continuous time model with state dependent service rates, followed by the second approximation step in Section 5. Section 6 gives the results for the stability analysis and Section 7 presents results from simulation to illustrate the accuracy of the model presented in this paper. Finally, Section 8 concludes the paper.

* Corresponding author.
E-mail address: t.j.m.coenen@utwente.nl (T. Coenen).

1434-8411/$ – see front matter © 2012 Published by Elsevier GmbH.
http://dx.doi.org/10.1016/j.aeue.2012.06.009

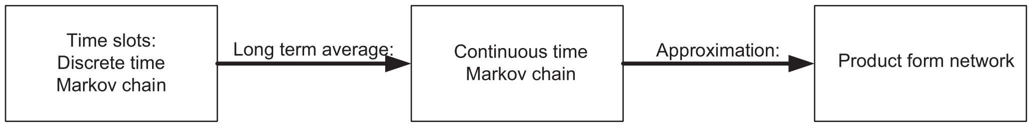

Fig. 1. Approximation steps.

2. Literature and contribution

The stability of networks, as considered in this paper, has received considerable interest in the literature. Also inspired by wireless networks, [1] analyzes a discrete time slotted ALOHA system. Bounds on the stability region are found using the concept of dominance. A different approach is presented in [2] where the rate stability and output rates are calculated for shared resource networks. Stability conditions for separate nodes are derived for general allocation functions under mild assumptions. The model discussed in this paper however does not fall under the set of allocation functions, as the overall capacity of the network is not constant. For a network of parallel servers with coupled service rates, necessary and sufficient conditions for stability are derived in [3]. Stability and performance of networks where the service rate depends on the network state is also analyzed in [4], where transmissions over links with a fixed capacity are considered. Opposed to the work presented in these papers, the rate allocated to a server does not depend on the number of packets present in the queue, but on the number of nodes competing.

Similar assumptions regarding the contention between nodes, where alive nodes block other nodes as discussed in this paper are made in [5]. The throughput in a multihop tandem network is considered both under saturation, where each node generates its own traffic and under a single flow over all nodes. The authors conjecture that a random access scheme severely degrades the throughput of the network.

Analytic results for a multihop network with two contending queues are presented in [6]. Using the theory of Riemann–Hilbert boundary value problems, the generating function of the stationary distribution is obtained. In [7] some performance measures of this system are analyzed, focussing on the computational issues that occur. Even for such a small network as considered in these papers, a complex analysis is needed to obtain analytical results. The approach we present is applicable for general size networks, however we do not obtain results on the stationary distribution, but on stability and throughput.

The optimal throughput a network can support, often referred to as the capacity of the network, is discussed in [8], which however does not focus on multi-hop networks. This aspect is addressed in [9], where for a single multi-hop flow a new capacity limit is derived. These results are limiting results for large networks. More detailed models are discussed in [10] for a tandem and lattice network with saturated nodes. They calculate the optimal offered load, preventing packet loss in a network with hidden nodes. This work is extended for multiple crossing flows in [11]. Instead of focussing on the specific parameters of the MAC protocol, as presented in these papers, we take a higher level view, providing valuable insights for general networks.

Next to limiting the capacity of a network, contention between nodes has an impact on the fairness of protocols, as in the equality in rate allocated to nodes or the throughput of flows. In [12] the authors describe the border effects in a CSMA/CA network and its impact on fairness. The stability and throughput for a weighted fair queueing model with saturated nodes is discussed in [13] showing that the throughput, while taking into account the topology, routing and random access in the MAC layer, in this setting does not depend on the load in the intermediate nodes as long as the network is stable.

Different aspects of importance for the stability and throughput of networks have also received much attention. Focussing on the impact of routing, [14] investigates the stability and throughput of static wireless networks with slotted time. The authors show that routing has a large impact on the stability properties and that as long as the intermediate queues in a network are stable, the throughput does not depend on the traffic generated at these intermediate nodes. In [15] the focus is on the calculation of the interference to noise ratio and show the influence of the network size and the data rate on this ratio and link this to the throughput of the network.

The contribution of this paper is that we provide a comprehensible model that very accurately predicts the bottlenecks and maximal throughput of a network, which also is applicable for networks with unstable nodes. The results provide insight in the impact that contention between nodes has on the performance of the network, without the need of a complex analysis.

3. Discrete time model

3.1. General model

Consider a network consisting of n queues with infinite buffers. Due to contention between nodes not all nodes can transmit their packets at the same time. We define the *contention set* $I(i)$, $i = 1, \ldots, n$, of a node i as the set of nodes blocked from transmission when node i is transmitting. A typical example is the set of nodes within a certain interference range. However, for the model there need not be a relation between the network structure and the contention set. The way contention between nodes takes place will be elaborated upon below. A set of J traffic flows $f(t_j)$, $j = 1, \ldots, J$, travel over multihop paths, denoted by the ordered sets t_j, from node $t_j(1)$ to $t_j(m_j)$, where we assume that no loops are made within a path, i.e. paths are simple and packets automatically follow their path. Traffic consists of equally sized packets that are transmitted one packet per time slot. An example of such a network is depicted in Fig. 2. In this example a network of 8 nodes is depicted. In the figure there are three flows: $f(t_1)$ from node 1 through node 2 to node 3 (i.e. we have that $t_1 = \{1, 2, 3\}$), $f(t_2)$ from node 1 through nodes 4 and 6 to node 8 and $f(t_3)$ from node 7 through nodes 6 and 5 to node 3.

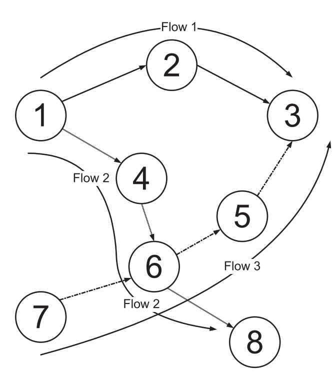

Fig. 2. Network with three traffic flows.

Packets arrive at the origin nodes $t_j(1)$ (i.e. nodes 1 and 7 in Fig. 2) according to a Poisson process with rate λ_j for flow j and are served first come first served. A node is called *stable* when its average service rate exceeds the average arrival rate of packets at the node, and a network is called stable when all its nodes are. A node that is unstable is called a *bottleneck* node. The average number of packets of a flow that reach the destination node per time unit is the throughput of this flow, which is limited by the service rate of the bottleneck nodes of the network. The main interest in this paper is the throughput of the flows and the identification of bottleneck nodes.

A node is called *alive* when it has packets to transmit and thus participates in the contention. In each time slot, all alive nodes contend to be allowed to transmit a packet. The probability of a node being allowed to transmit depends on the set of nodes contending, which we focus on in the following section. In each time slot a node is either not contending, blocked or allowed to transmit. In each time slot this process is repeated, where we assume the selection of nodes being allowed to transmit to be independent between time slots.

Let p_i denote the probability that node i is alive and let π be a liveliness vector, such that $\pi_i = 1$ if node i is alive and $\pi_i = 0$ otherwise. The set of all 2^n possible liveliness vectors is denoted by Π. The probability that a liveliness vector π occurs is denoted by q_π. The probability that node i transmits under liveliness vector π is denoted by $r_{i,\pi}$. The network can be represented by a discrete time Markov chain with the queue lengths at each node as the state of the system. Actually, as packets are forwarded to a next node depending on the flow they belong to, also the type of the packets in the queue needs to be included in the state description. However, in our steady state description these types will not play a role and are therefore omitted from the state description. The transition probabilities depend on the state of the system via the liveliness vector only, i.e. the number or type of packets in a queue does not affect the probability of a node transmitting in a slot, unless it is empty.

3.2. Contention

Multiple nodes can only be transmitting simultaneously in the same time slot when they are outside of each others contention set. If multiple nodes within each others contention set are alive, the contention protocol decides which nodes may transmit. The probability that a node is allowed to transmit a packet in the following slot can be determined when the contention sets, the protocol in use and the competing nodes are known. We assume an ideal contention protocol, where no collisions will occur and hence no packets will be lost.

As we are not interested in the details of the contention protocol but only the corresponding probabilities for nodes to transmit, we will use a simple protocol giving each node an initial equal probability of winning a contention. For other protocols, transmission probabilities can also be calculated. Using

$$|\pi| = \sum_{i=1}^{n} \pi_i, \tag{1}$$

i.e. $|\pi|$ equals the number of alive nodes and (taking e_m as the unit vector of length n, with all zeros except a 1 on location m),

$$\tilde{\pi}(k) = \pi - \sum_{m \in I(k):\pi_m=1} e_m \tag{2}$$

as the liveliness vector remaining after a node k blocks all nodes in its contention set (as it won the contention), the probability $r_{i,\pi}$ that a node i may transmit a packet under liveliness vector π can be calculated using the following recursion:

$$r_{i,\pi} = \begin{cases} 0 & \text{for } \pi_i = 0 \\ (1 + \sum_{k \neq i:\pi_k=1} r_{i,\tilde{\pi}(k)})/|\pi| & \text{for } \pi_i = 1 \end{cases} \tag{3}$$

with $r_{i,\mathbf{0}} = 0$, where $\mathbf{0}$ denotes a liveliness vector with no alive nodes. This can be seen as follows: with equal probability of $1/|\pi|$ any non-empty node (so node i itself or any other alive node k) wins the direct contention. Assuming node k wins the contention, it blocks all nodes in its contention region, reducing the liveliness state to $\tilde{\pi}(k)$, after which all remaining nodes contend again. Any node that did not win the contention, but was not blocked hence can compete again and might win the new contention, with probability $1/|\tilde{\pi}(k)|$. This process continues until all non-empty nodes either are allowed to transmit or are blocked.

As an example, consider the network as depicted in Fig. 2 with contention sets chosen such that nearby nodes contend: $I(1) = \{2, 4\}$, $I(2) = \{1\}$, $I(4) = \{1, 5, 6\}$, $I(5) = \{4, 6\}$, $I(6) = \{4, 5, 7\}$, $I(7) = \{6\}$. Assume that all 6 nodes have packets to transmit (as nodes 3 and 8 do not transmit packets they are never alive), so that $\pi = (1, 1, 0, 1, 1, 1, 1, 0)$. The probability $r_{4,\pi}$ that node 4 will be allowed to transmit by directly winning the contention is $1/|\pi| = 1/6$. If for example node 1 wins the contention, node 4 is blocked, as it is in its contention set. As $\tilde{\pi}(1) = (0, 0, 0, 0, 1, 1, 1, 0)$, we get $r_{4,\tilde{\pi}(1)} = 0$ as $\tilde{\pi}_4(1) = 0$. The same holds if node 5 or 6 wins the contention, as $r_{4,\tilde{\pi}(5)} = 0$ and $r_{4,\tilde{\pi}(6)} = 0$. If node 2 or 7 wins the contention, node 4 still could be allowed to transmit. The probability that node 4 wins contention after node 2 has won the contention is given by $r_{4,\tilde{\pi}(2)}$, where $\tilde{\pi}(2) = (0, 0, 0, 1, 1, 1, 1, 0)$. This probability can be calculated by calculating $r_{4,\pi}$, but with $\pi = (0, 0, 0, 1, 1, 1, 1, 0)$, showing the recursion. As after each step, but with the new value of π, the number of zeroes in the liveliness vector increases, the recursion will stop when $\pi = (0, 0, 0, 0, 0, 0, 0, 0)$. For this example, the probability the nodes may transmit are given by [19/48, 29/48, 0, 14/48, 20/48, 19/72, 53/72, 0]. We further analyze this network in Section 7.2.

Note that the overall probability of being allowed to transmit is not equal for all nodes. A similar analysis to obtain $r_{i,\pi}$ can be done for any network, with any contention sets and protocol. More extensive calculations will be needed for larger networks with different topologies, but the principle will not change. In the remainder of this paper, we will assume that the contention regions and protocol are known, such that all conditional rates $r_{i,\pi}$ of the nodes can be calculated.

4. Approximation step 1: continuous time

We are interested in the long term average behavior of the network, especially the throughput and stability issues. Considering the system on a higher level and a larger time scale, the discrete character due to the time slots fades and the model can be seen as a continuous time Markov process. The state of the system consists of the number and type of packets at each queue, but as the state of the system only influences the transition rates through the liveliness of the network, we do not focus on the queue lengths. The flow a packet being served belongs to determines the direction in which it will be forwarded. We incorporate this into the model as described below. In the following we will denote parameters used for the continuous time approximation by adding a hat to the equivalent parameter in the original discrete time model.

When a queue has packets available and the liveliness is given by π, the probability of a packet being sent is given by $r_{i,\pi}$. On average, the number of packets sent per slot under state π hence is $r_{i,\pi}$. For the continuous time Markov chain, we approximate the service rate under state π of the node using the exponential distribution with rate $\hat{r}_{i,\pi} = r_{i,\pi}$. The probability of a node being alive or not depends on the arrival rate of packets and the service rate at the node. We first focus on the arrival rate of packets.

Whenever the nodes on a multihop path t_j preceding a node $t_j(i)$ are stable, the arrival rate from this flow will be λ_j, the external arrival rate of the flow. The total arrival rate of traffic a_i at node i is given by

$$a_i = \sum_{j:i\in t_j} \lambda_j(i) \tag{4}$$

where $\lambda_j(i)$ is the arrival rate at node i for flow $f(t_j)$. When the network is stable, this simplifies to $a_i = \sum_{j:i\in t_j} \lambda_j$. When there are unstable nodes in the network, the arrival rate of packets at each queue can be determined as follows. Due to the multihop feed forward structure of the network we have that the arrival rate $\lambda_j(i)$ is determined by its preceding nodes. If one or more of the preceding nodes are unstable, the average arrival rate for the nodes after the bottleneck on this path will depend on the service rate of the unstable nodes. The probability $p_{t_j(i-1)t_j(i)}$ that a served packet at node $t_j(i-1)$ continues to node $t_j(i)$, the packet is of flow $f(t_j)$, is given by

$$p_{t_j(i-1)t_j(i)} = \frac{\lambda_j(t_j(i-1))}{a_{t_j(i-1)}}. \tag{5}$$

The arrival rate $\lambda_j(t_j(i))$ from flow $f(t_j)$ at node $t_j(i)$ is given by

$$\lambda_j(t_j(i)) = \min(\lambda_j(t_j(i-1)), p_{t_j(i-1)t_j(i)}\hat{r}_{t_j(i-1)}), \tag{6}$$

where $\lambda_j(t_j(1)) = \lambda_j$, the external arrival rate of packets at the first node in path t_j. This can be seen as follows: either the preceding node can serve all its incoming traffic, or its service rate is too low. In the latter case, the fraction of the service rate of node $t_j(i-1)$ that is used for flow $f(t_j)$, equal to $p_{t_j(i-1)t_j(i)}$, determines the arrival rate at the next node for this flow. Here $\hat{r}_{t_j(i-1)}$ denotes the average state independent service rate of node $t_j(i-1)$, which will be determined in the next section. Assuming this rate is known, Eqs. (5) and (6) give a system of equations that can easily be solved, giving the arrival rate per flow at each node. We use these arrival rates in the analysis of the liveliness of the system, which influences the service rate of the nodes.

5. Approximation step 2: product form network

The Markov chain with state dependent service rates is not amenible for analysis. For a network with only two queues in tandem, this equals the model presented in [6] under deterministic service times. Even for such a small network, a complex analysis is needed to obtain analytical results. Therefore, for an arbitrary network, we approximate the continuous time approximation by obtaining an appropriate *state independent* service rate for each node to analyze the behavior of the network.

The state independent service rate $\hat{r}_i$ is obtained by considering the long term average percentage of time the system is in a state with liveliness vector π. The probability of node i being alive is given by

$$\hat{p}_i = \min(\frac{a_i}{\hat{r}_i}, 1). \tag{7}$$

For the final approximation step, let $\hat{q}_\pi$ denote the steady state probability that the liveliness vector is π (to be calculated later) and assume the state independent average service rate of a node i in the network to be given by,

$$\hat{r}_i = \sum_{\pi\in\Pi} \frac{\hat{r}_{i,\pi}\hat{q}_\pi}{\hat{p}_i} \tag{8}$$

We obtain Eq. (8) by considering a large time scale and weighing the service rate over the possible liveliness of the system, i.e. by unconditioning on the liveliness, but conditioning on the node being alive. The state independent service rate $\hat{r}_i$ can be seen as the average rate at which a node services packets, given that it is alive.

Theorem 1. *The steady state probability $\hat{q}_\pi$ that the system is in a state with liveliness vector π is given by*

$$\hat{q}_\pi = \prod_{i=1}^{n} (1-\hat{p}_i)^{(1-\pi_i)}\hat{p}_i^{\pi_i}. \tag{9}$$

□

Proof. Summarizing the above, we have the following assumptions for the state independent continuous time approximation:

1. The external arrival process of traffic at queues is a Poisson process.
2. There is infinite waiting space at all the queues.
3. The service time at the queues has an exponential distribution and is independent of the state of the system and arrival process.
4. After completion of service at queue i a packet instantaneously moves to the next queue k with probability p_{ik}, $k = 1, \ldots, n$, for additional service or with probability p_{i0} the packet completes service and leaves the system, where we have that $\sum_{k=0}^{n} p_{ik} = 1$. The routing probabilities are independent of the history of the system.

A network for which the assumptions (1)–(4) hold is a product form network (c.f. [16]). Hence, the probability of a certain state of the system occurring is the product of the probabilities of nodes containing a certain number of packets. As the state of the system directly implies a certain liveliness, also the liveliness vector can be found as the product of the liveliness of separate nodes, showing (9) holds. □

We will now use Eqs. (4)–(9) as an approximation for the discrete time model. This rather coarse approximation will provide quite accurate results, as we are interested in the influence of the load on *average* behavior of the network.

6. Stability

The average service rate $\hat{r}_i$ at which each node operates determines the load under which the network is stable. As presented earlier, the average service rate of a node i is given by (8) and the probability that a node is alive by (7). Writing out the expression for $\hat{q}_\pi$ and inserting (7) into (8), we obtain n equations, with $2n$ unknowns, which are the a_i and the $\hat{r}_i$. Assuming that all nodes are stable, that is when all $a_i < \hat{r}_i$, the arrival rate at each node is known. The values for $\hat{r}_i$ can hence be calculated for a stable system. However, it is still to be determined for which values of λ_j (and thus a_i) the network is stable.

As presented in Section 4, the arrival rate for a certain flow j at node $t_j(i)$ is given by (6) and the total arrival rate by (4). Using the n Eq. (4) for a_i, it is possible to solve the system of $2n$ unknown variables, which entails solving polynomials of degrees that increase exponentially with the network size. Solutions can be obtained numerically, however, using for instance the following algorithm to obtain the values of $\hat{r}_i$.

Algorithm 2. Calculation of $\hat{r}_i$, $i = 1, \ldots, n$

1. Set all values $\hat{r}_i$ to 1, $i = 1, \ldots, n$
2. Calculate $\lambda_j(k)$, $j = 1, \ldots, J$ and $k = t_j(1), \ldots, t_j(m_k)$
3. Calculate $a_i = \sum_{j:i \in t_j} \lambda_j(i)$, $i = 1, \ldots, n$
4. Calculate $\hat{p}_i = min(a_i/\hat{r}_i, 1)$, $i = 1, \ldots, n$
5. Calculate $\hat{q}_\pi = \prod_{i=1}^{n} (1 - \hat{p}_i)^{(1-\pi_i)} \hat{p}_i^{\pi_i}$, $\pi \in \Pi$
6. Calculate new $\hat{r}_i = \sum_{\pi \in \Pi} (\hat{r}_{i,\pi} \hat{q}_\pi / \hat{p}_i)$, $i = 1, \ldots, n$
7. Calculate the difference $\epsilon_i = \hat{r}_i(new) - \hat{r}_i(old)$, $i = 1, \ldots, n$
8. Repeat step 2 till 7 until convergence occurs, that is $|\epsilon| \le \delta$ for an appropriate value of δ.

□

We have numerically established that Algorithm 2 converges to a unique solution $\hat{r}_i$ for any values of λ_j, $j = 1, \ldots, J$. For an analysis of the algorithm we refer Appendix A.

Using Algorithm 2, the service rate of all nodes can be calculated for any set of flows through the network. The corresponding arrival rates at the destination nodes of the flows give the throughput of the network. Whenever the network is stable, the total throughput will equal $\sum_j \lambda_j$. For a general network, the calculation of the throughput, independent of the topology of the network, involves solving n equations in n unknowns. Using Algorithm 2, the arrival rate(s) can be chosen arbitrarily. To determine the stability range of the network, we separately consider each flow in the network. Fixing the arrival rates of all but one flow (such that the system with these flows is stable), there exists a value λ_{opt} for the remaining flow such that $a_k = \hat{r}_k$ for at least one $k \in 1, \ldots, n$, which provides the maximal throughput λ_{opt} of this flow. Node k is then the bottleneck of the network. In this manner the stability range of the network can be calculated (examples are shown in the following section).

7. Examples and validation

7.1. Multihop tandem network

In the following we analyze a multihop tandem network. When considering a general network, the analysis of the stability region involves considering flows separately. First, we show how for a specific contention protocol the transmission probabilities $r_{i,\pi}$ can be calculated in this network, which corresponds to a single multihop transmission in a network. Next, we use simulation to validate results obtained by our algorithm for different sizes of the network. Some surprising results are obtained, which are correctly predicted by our model.

Consider a tandem network of size n. The average service rate at which a node transmits depends on the position in the tandem network. As indirectly all nodes in the network influence each other, the total length of the network has an impact. This impact when all nodes are alive is shown, using a contention protocol selecting a node to transmit with equal probability among all alive nodes.

Consider the tandem network such that nodes cannot transmit and receive at the same time. A node that is allowed to transmit hence blocks its direct neighbor(s). When all n nodes are alive, each node has a probability $1/n$ of obtaining the channel directly and blocking its neighbor(s). The remaining nodes continue contending for the channel until they are either blocked or allowed to transmit. The rate $r_{i,\mathbf{1}}(n)$ for a node at position i in a fully alive tandem network of length n can be calculated using

$$r_{i,\mathbf{1}}(n) = \frac{1}{n}\left[\sum_{k=1}^{i-2} r_{i-k-1,\mathbf{1}}(n-k-1) + 1 + \sum_{k=i+2}^{n} r_{i,\mathbf{1}}(k-2)\right]. \quad (10)$$

The right hand side of (10) follows from the node winning the contention: if the first node in the network wins the contention, it blocks the second node and the remaining $n-2$ nodes compete, with node i now at position $i-2$. Otherwise, in a similar manner, a node before (but not a neighboring) node i wins the contention, node i wins the contention itself, either of node i's neighbors wins the contention or a node k behind node i wins the contention. Each of these events occurs with a probability of $1/n$, together giving the recursive formula.

Note that a multihop tandem network (in this setting) with nodes that are not alive can be decomposed into many smaller multihop networks. For a fully alive tandem network where nodes cannot transmit and receive at the same time, Table 1 shows the rates for different lengths of the network.

Theorem 3. *For the multihop tandem network with all alive nodes, the rate allocated to the nodes converges when the network size increases, where in particular*

$$\lim_{n\to\infty} r_{1,\mathbf{1}}(n) = 1 - \frac{1}{e} \quad \text{and} \quad \lim_{n\to\infty} r_{2,\mathbf{1}}(n) = \frac{1}{e}. \quad (11)$$

□

Proof. For the proof we refer to Appendix A.2. □

Other limits are observed in Table 1, showing that the border effects fade for the middle nodes as the length of the network increases, in accordance with [12]. This border effect already starts to fade for networks of size 12.

We note that the calculation of the rates $r_{i,\pi}$ for the linear setting has the pleasant property that the rate of a certain node i under liveliness π is only dependent on the number of nodes that are alive and directly connected to each other. When considering different

Table 1
Transmission probability for a fully alive tandem network.

Size/node	1	2	3	4	5	6	7	8	9	10	11	12
1	1	–	–	–	–	–	–	–	–	–	–	–
2	0.5	0.5	–	–	–	–	–	–	–	–	–	–
3	0.6666	0.3333	0.666	–	–	–	–	–	–	–	–	–
4	0.625	0.375	0.375	0.625	–	–	–	–	–	–	–	–
5	0.6333	0.3667	0.4667	0.3667	0.6333	–	–	–	–	–	–	–
6	0.6319	0.3681	0.4444	0.4444	0.3681	0.6319	–	–	–	–	–	–
7	0.6321	0.3679	0.4488	0.4262	0.4488	0.3679	0.6321	–	–	–	–	–
8	0.6321	0.3679	0.4481	0.4297	0.4297	0.4481	0.3679	0.6321	-	–	–	–
9	0.6321	0.3679	0.4482	0.4291	0.4334	0.4291	0.4482	0.3679	0.6321	–	–	–
10	0.6321	0.3679	0.4482	0.4292	0.4328	0.4328	0.4292	0.4482	0.3679	0.6321	–	–
11	0.6321	0.3679	0.4482	0.4292	0.4329	0.4322	0.4329	0.4292	0.4482	0.3679	0.6321	–
12	0.6321	0.3679	0.4482	0.4292	0.4329	0.4323	0.4323	0.4392	0.4292	0.4482	0.3679	0.6321

Please cite this article in press as: Coenen T, et al. Bottlenecks and stability in networks with contending nodes. Int J Electron Commun (AEÜ) (2012), http://dx.doi.org/10.1016/j.aeue.2012.06.009

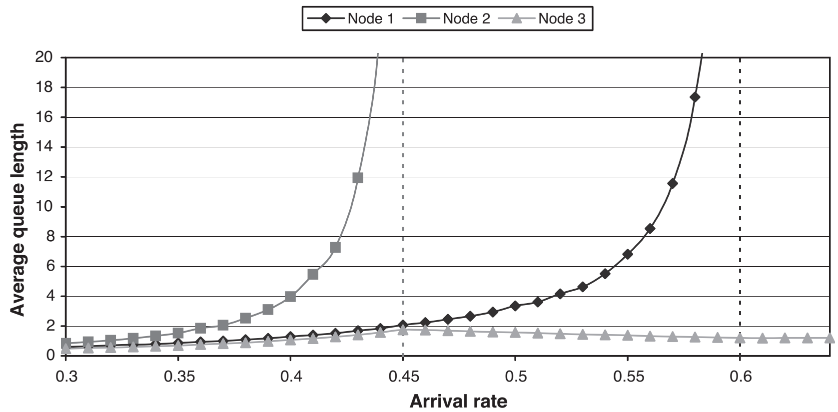

Fig. 3. Simulated average queue length in a 3 hop network with the instability rates calculated by the model.

contention sets and protocol or network layout, this property however may no longer be present.

To validate the results presented in this paper, a simulation model has been constructed that mimics the behavior of the discrete time network under consideration. The arrival and processing of the packets is modeled, with a simulation for each parameter setting lasting one million simulated time slots after a warm up period of 100.000 slots. The results are compared with the stability ranges and the throughput of the network calculated with the state independent continuous time approximation, using the provided algorithm. For some settings, we provide the exact derivation of the results.

Consider the multihop tandem network for $n = 3$. The average service rates at which the nodes operate are given by (using (8) and (9))

$$\hat{r}_1 = (1-\hat{p}_2) + \frac{1}{2}\hat{p}_2(1-\hat{p}_3) + \frac{2}{3}\hat{p}_2\hat{p}_3 \tag{12}$$

$$\hat{r}_2 = (1-\hat{p}_1)(1-\hat{p}_3) + \frac{1}{2}\hat{p}_1(1-\hat{p}_3) + \frac{1}{2}(1-\hat{p}_1)\hat{p}_3 + \frac{1}{3}\hat{p}_1\hat{p}_3$$

$$\hat{r}_3 = (1-\hat{p}_2) + \frac{1}{2}(1-\hat{p}_1)\hat{p}_2 + \frac{2}{3}\hat{p}_1\hat{p}_2.$$

Obviously, the second node will be the bottleneck of the network as $\hat{r}_2$ is smaller than $\hat{r}_1$ and $\hat{r}_3$, as it is the only node contending with two neighbors. When node 2 is unstable, we have that $\hat{p}_2 = 1$. To determine at what arrival rate λ this will occur, we use that $\lambda = a_i = \hat{r}_2$, so that

$$\hat{p}_1 = \frac{\hat{r}_2}{\hat{r}_1} \quad \text{and} \quad \hat{p}_3 = \frac{\hat{r}_2}{\hat{r}_3}. \tag{13}$$

Combining Eqs. (12) and (13) with $\hat{p}_2 = 1$ we find that $\hat{p}_1 = \hat{p}_3 = (9-\sqrt{57})/2$, resulting in the critical arrival rate of $\lambda = \hat{r}_2 = 8 - \sqrt{57}$. From this value of λ on the second node will be unstable. If we increase the arrival rate even more, the first node will also become unstable. The third node however will always remain stable, as its service rate will always be higher than the service rate at the second node, which determines the arrival rate at the third node. To find from which value of λ on the first node will also be unstable, we substitute $\hat{p}_1 = \hat{p}_2 = 1$ in (12) which leads to $\hat{p}_3 = 0.6$, and the rate at which node 1 becomes unstable equals $\lambda = \hat{r}_1 = 0.6$. Also note that the rate of the second node has now fallen to a value of $\hat{r}_2 = 0.4$, so that the throughput of the network has decreased.

For the three node tandem network, Fig. 3 shows the average queue length at the three nodes for increasing load of the system and Fig. 4 shows the throughput of the system. The calculated values of arrival rates for which queues become unstable are depicted as dotted vertical lines in the figures.

As can be seen in Figs. 3 and 4, the arrival rates at which the first and second node become unstable coincide with the calculated values. Additional simulations for the arrival rates near the ones causing instability of nodes were performed to confirm the results, but are not shown in the figures to maintain readability. The throughput, which reaches a maximum of $8-\sqrt{57} \approx 0.4501$ when the second node becomes unstable, decreases after this value. This decrease in throughput is caused by the decrease in service rate at the second node, as the first node becomes more highly loaded. This causes the first queue to be alive a larger fraction of the time, blocking the second node. The throughput settles at 0.4 after the first node has become unstable at an arrival rate of 0.6, which is in agreement with the values calculated.

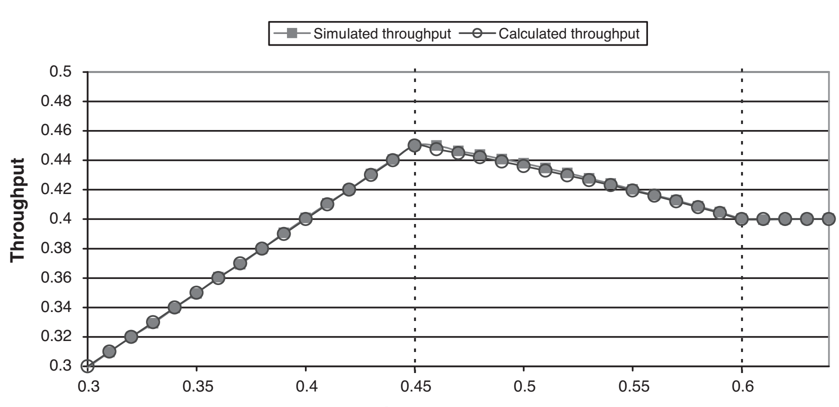

Fig. 4. Simulated and calculated throughput of a 3 hop network with the instability rates calculated by the model.

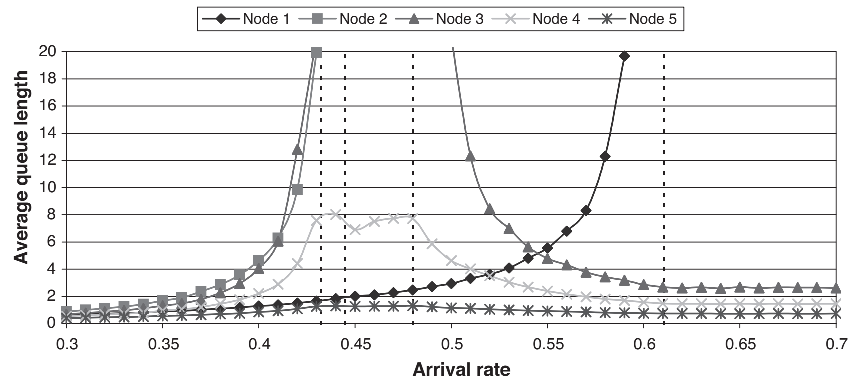

Fig. 5. Simulated average queue length in a 5 hop network with the instability rates calculated by the model.

Next, considering a larger network with 5 hops, one might expect that it is the second node that becomes the bottleneck. Using the presented model and setting $\hat{p}_2 = 1$ however shows that no real valued solution exists, meaning that node 2 cannot be the node to become unstable first. It actually is the third node that becomes the bottleneck first at an arrival rate of 0.4323, which is the maximum throughput of the network. Increasing the arrival rate to 0.4448 causes the second node to become unstable as well. Increasing the arrival rate further, the third node becomes stable again. The presented model also determines the arrival rate at which this occurs by making a small adjustment to the equations. As the third queue will become stable as soon as its average service rate is lower than the second queue's rate, we now set $\hat{r}_2 = \hat{r}_3$. As both queues are still unstable we have that $\hat{p}_2 = \hat{p}_3 = 1$ and that $\hat{p}_4 = \hat{r}_2/\hat{r}_4$ and $\hat{p}_5 = \hat{r}_2/\hat{r}_5$. Using the standard equations for the $\hat{r}_i$'s and setting $\hat{p}_1 = \lambda/\hat{r}_1$, we solve the system to obtain $\lambda = 0.4803$ and $\hat{r}_2 = \hat{r}_3 = 0.4306$. Finally increasing the arrival rate to 0.6108 causes the first node to become unstable, resulting in a throughput of 0.3892. Simulation of the network under consideration provided the results as presented in Figs. 5 and 6 where the vertical lines show the calculated values for which nodes become (un)stable.

That it is queue 3 that is the first node to become unstable can be called surprising. When all queues are alive, the average service rate of queue 2 is lower than that of queue 3. However, when queue 1 and/or queue 5 are empty, the third queue has the lowest rate (see Table 1 for a 3–5 node network). As can be seen in Fig. 5, the average queue length at nodes 1 and 5 are low for the load when queue 2 and 3 are already reaching instability. This indicates that they frequently will not be alive, which is in the disadvantage of the third node, making it the bottleneck node. However, as the arrival rate increases, nodes 1 and 5 will be alive more often, which is beneficial for node 3, resulting in the queue becoming stable again. Surprising as this behavior may be, it is predicted correctly by the model.

7.2. General eight node network

Consider the network as depicted in Fig. 2. Note that any set of interfering nodes can be used, mimicking the behavior of any access control protocol, i.e. to mimic an RTS/CTS protocol all nodes within transmission range of the sending and receiving node can be used as the contention set. To avoid trivial results we set the interference ranges for this example to be (only showing the nodes that need to transmit) $I(1) = \{2, 4\}$, $I(2) = \{1\}$, $I(4) = \{1, 5, 6\}$, $I(5) = \{4, 6\}$, $I(6) = \{4, 5, 7\}$, $I(7) = \{6\}$. First flow $f(t_1)$ is set up, with rate $\lambda_1 = 0.1$. Obviously the network can handle this flow. Second, flow $f(t_3)$ is set up, with rate $\lambda_3 = 0.1$ as well. Again, the network remains stable (note that even though both flows have node 3 as endpoint, this does not cause problems as we assume perfect reception of all transmissions). Now flow $f(t_2)$ is initiated and the open question is which rate can be achieved for this flow. The arrival rates of traffic at the nodes, as long as the network is stable, is given by

Node	1	2	3	4
Arrivalrate	$\lambda_2 + 0.1$	0.1	0.2	λ_2

Node	5	6	7	8
Arrivalrate	0.1	$\lambda_2 + 0.1$	0.1	λ_2

and the probabilities q_π of all possible liveliness vectors can easily be calculated. Using these values in Eqs. (7) and (8) gives a set of 8 equations with 9 unknowns (all the $\hat{r}_i$ and λ_2), which can be solved when it is known which node becomes the bottleneck. Using $\lambda_2 = \hat{r}_i$

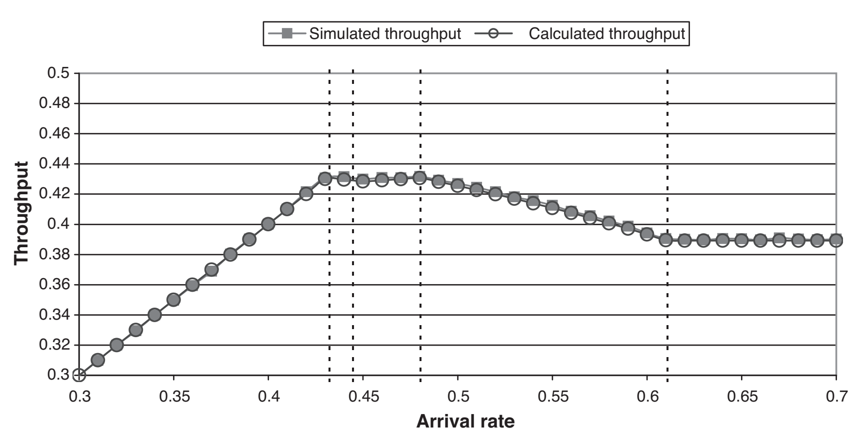

Fig. 6. Simulated and calculated throughput of a 5 hop network with the instability rates calculated by the model.

Please cite this article in press as: Coenen T, et al. Bottlenecks and stability in networks with contending nodes. Int J Electron Commun (AEÜ) (2012), http://dx.doi.org/10.1016/j.aeue.2012.06.009

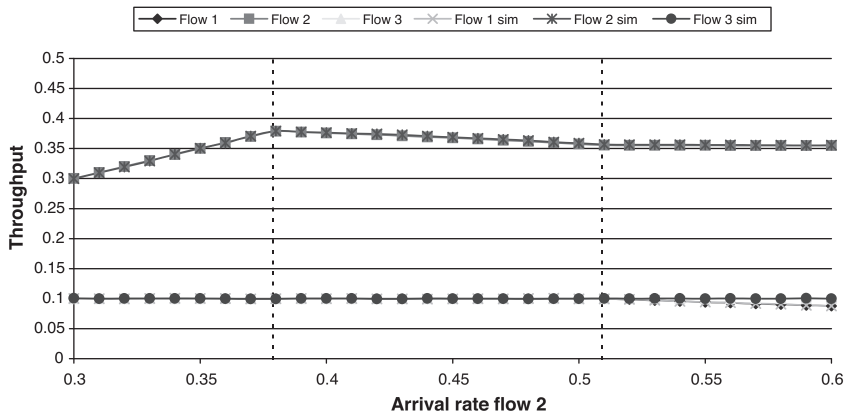

Fig. 7. Simulated and calculated throughput in the 8 node network with the instability rates calculated by the model.

for $i = 1, 4, 6$ and solving shows that it is node 4 that becomes the bottleneck at an arrival rate of $\lambda_2 = 0.3789$. Increasing the arrival rate λ_2 further causes node 1 to become unstable as well, influencing the throughput of the first flow. Using the model, this is calculated to happen at an arrival rate of $\lambda_2 = 0.5092$. Fig. 7 shows the throughput of the separate flows for an increasing arrival rate of the second flow. Both the values calculated by the model and the simulation results are shown.

Numerical evaluation shows that the model gives very accurate predictions of the throughput, where the error at each calculated point stays below 1%. The load at which node 1 and 4 become unstable can be recognized as the points where the slope of the graph changes, where the simulation again shows that this is at the point predicted by the model.

8. Conclusion

Inspired by wireless ad hoc networks, where interference limits the capacity, networks with contending nodes are analyzed in this paper. Each time slot, nodes compete to transmit a packet from their queue, where a winning node blocks other nodes in its neighborhood. The time slot system is approximated in two steps. First, by considering the long run average behavior of the discrete time system, a continuous time model is obtained. As the second step, appropriate state independent service rates for the nodes in the network are determined. Combining relations between the arrival and service rates of the nodes, bottleneck nodes are identified which determine the throughput of a multihop wireless network. Using the two rather coarse approximation steps, we propose a product form network approximation. Taking advantage of the properties of product form networks, equations for the liveliness vector (whether nodes have packets in their queues or not) and the average service rates of the nodes are derived and solved using a simple algorithm. Surprisingly, the continuous approximation for the long term average behavior turns out to give accurate results concerning the stability and throughput of the network. Other performance measures, as the queue length and waiting time, have not been considered.

Our approach provides very accurate results for the lowest arrival rate of a flow at which one of the nodes becomes unstable, thus giving the maximal throughput for this flow. Also, increasing the arrival rate further, instability of the rest of the nodes is analyzed. Our model correctly predicts surprising behavior in a multihop tandem network, where a queue at first turning out to be the bottleneck, returned to stability again after increasing the arrival rate. The approach presented is applicable for general networks, with various contention settings and protocols. Using simulations of the discrete time system, results were compared with the continuous time model, showing that the model provides very accurate results.

Appendix A.

A.1. Analysis of Algorithm 2

To analyze the convergence of Algorithm 2, we consider the separate steps and the recursion. The initial value of $\hat{r}_i = 1$ corresponds to a network without contention, immediately giving an indication whether the network is stable or not. To calculate all $\lambda_j(k)$'s in step 2, Eqs. (6), (4) and (5) need to be combined, giving Jm equations with equally many unknown variables which can be solved. From these values, obviously steps 3–6 can be calculated, leading to the recursion.

Let $g(r)$ denote the function that calculates the new value of r using the steps described. The function $g(\cdot) : \mathbb{R}^n \to \mathbb{R}^n$ is a continuous function on the convex compact subset $[0, 1]^n$. Following Brouwers fixed point theorem (c.f. [17]), we consider the equation $g(r) = r$, which has a solution, which we need to show to be the unique fixed point. To achieve this, we use the Contraction Mapping Theorem (CMT, c.f. [17]), saying that the equation $g(r) = r$ has a unique solution if and only if

- The function $g(\cdot)$ maps $[0, 1]^n$ to $[0, 1]^n$
- There is a constant $G < 1$ such that $||g(x) - g(y)|| \leq G||x - y||$ for all $x, y \in [0, 1]^n$

First, the algorithm needs to be shown to map any starting value for r to another value of r that is within the possible range of $[0, 1]^n$. For this to be the case, we need that

$$0 \leq \sum_{\pi} \hat{r}_{i,\pi} \hat{q}_\pi \leq \hat{p}_i.$$

The first inequality is obvious, for the second one we note that $\hat{r}_{i,\pi} = 0$ for all π such that $\pi_i = 0$ and that $\hat{r}_{i,\pi} \leq 1$. This gives that

$$\begin{aligned}\sum_{\pi} \hat{r}_{i,\pi} \hat{q}_\pi &\leq \sum_{\pi:\pi_i=1} \hat{q}_\pi \\ &= \sum_{\pi:\pi_i=1} \prod_{j=1}^{n} (1 - \hat{p}_j)^{1-\pi_j} \hat{p}_j^{\pi_j} \\ &= \hat{p}_i \sum_{\pi:\pi_i=1} \prod_{j \neq i} (1 - \hat{p}_j)^{1-\pi_j} \hat{p}_j^{\pi_j} = \hat{p}_i,\end{aligned}$$

where the last equality holds as we sum over all possible liveliness states for the network without node i, proving the first part of the contraction mapping theorem.

The second part is more involved. We provide a complete proof for a two node network and indicate why the second condition is conjectured to hold for larger networks.

When following the steps of the algorithm for a two node tandem network, we have that

$$a(1) = \lambda(1) = \lambda \quad \text{and} \quad a(2) = \lambda(2) = \min(\lambda, \hat{r}_1)$$

$$\hat{p}_1 = \min\left(\frac{\lambda}{\hat{r}_1}, 1\right) \quad \text{and} \quad \hat{p}_2 = \min\left(\frac{\min(\lambda, \hat{r}_1)}{\hat{r}_2}, 1\right)$$

$$\hat{r}_1 = 1 - \frac{1}{2}\hat{p}_2 \quad \text{and} \quad \hat{r}_2 = 1 - \frac{1}{2}\hat{p}_1.$$

Note that as $0 \le \hat{p}_i \le 1$ we have that $\hat{r}_i \in [(1/2), 1]$. First assuming we are dealing with a stable network, the arrival rate at both nodes equals λ. We now (by substituting p_i) have the functional vector

$$g(\hat{r}) = \left(1 - \frac{\lambda}{2\hat{r}_2}, 1 - \frac{\lambda}{2\hat{r}_1}\right).$$

This gives, for $x = (x_1, \ldots, x_n)$,

$$||g(x) - g(y)||^2 = \left(\frac{\lambda}{2x_2y_2}\right)^2 (x_2 - y_2)^2 + \left(\frac{\lambda}{2x_1y_1}\right)^2 (x_1 - y_1)^2,$$

and for this to be smaller than $||x - y||^2$ we need to have that $(\lambda/2x_iy_i)^2 < 1$. As we assumed a stable network, we have that $\lambda < x_i$, so that

$$\frac{\lambda}{2x_2y_2} < \frac{1}{2y_i} \le 1$$

since $y_i \in [(1/2), 1]$ and so indeed the second condition holds proving that for a stable system the algorithm converges. If the system would be unstable, we have that

$$g(\hat{r}) = \left(1 - \frac{\min((\min(\lambda, \hat{r}_1)/\hat{r}_2), 1)}{2}, 1 - \frac{\min((\lambda/\hat{r}_1), 1)}{2}\right),$$

where the following situations can occur: $\lambda \ge \hat{r}_1$ or $\hat{r}_2 \le \lambda < \hat{r}_1$. In the first case we have that

$$g(\hat{r}) = \left(1 - \frac{1}{2}\min\left(\frac{\hat{r}_1}{\hat{r}_2}, 1\right), \frac{1}{2}\right)$$

which within two steps of the algorithm leads to $g(\hat{r}) = (1/2, 1/2)$ and thus converges to this unique solution. In the second case we have that

$$g(\hat{r}) = \left(\frac{1}{2}, 1 - \frac{\lambda}{2\hat{r}_1}\right),$$

$$||g(x) - g(y)||^2 = \left(\frac{\lambda}{2x_1y_1}\right)^2 (x_1 - y_1)^2$$

and $(\lambda/2x_1y_1)^2 < 1$ as shown earlier completing the proof that the algorithm converges for this two node network.

Considering a three node network, we obtain the following function (omitting the hat in the notation):

$$\begin{aligned} g(r) = \Bigg(& 1 - \frac{\min((\min(\lambda, r_1)/r_2), 1)}{2} \\ &+ \frac{\min((\min(\lambda, r_1)/r_2), 1)\min((\min(\min(\lambda, r_1), r_2)/r_3), 1)}{6}, \\ & 1 - \frac{\min((\lambda/r_1), 1)}{2} - \frac{\min((\min(\min(\lambda, r_1), r_2)/r_3), 1)}{2} \\ &+ \frac{\min((\lambda/r_1), 1)\min((\min(\min(\lambda, r_1), r_2)/r_3, 1)}{)} 3, \\ & 1 - \frac{\min((\min(\lambda, r_1)/r_2), 1)}{2} + \frac{\min((\lambda/r_1), 1)\min((\min(\lambda, r_1)/r_2), 1)}{6}\Bigg). \end{aligned}$$

As we have that $g(p) = (1 - (p_2/2) + (p_2p_3/6), 1 - (p_1/2) - (p_3/2) + (p_1p_3/3), 1 - (p_2/2) + (p_1p_2/6))$, starting in $([(1/2), 1], [(1/3), 1], [(1/2), 1])$, $g(\cdot)$ will also project on this range. For the CMT to hold, we first consider the stable system again, so that $\lambda < r_i$. In this case we have that

$$g(r) = \left(1 - \frac{1}{2}\frac{\lambda}{r_2} + \frac{1}{6}\frac{\lambda^2}{r_2r_3}, 1 - \frac{1}{2}\frac{\lambda}{r_1} - \frac{1}{2}\frac{\lambda}{r_3} + \frac{1}{3}\frac{\lambda^2}{r_1r_3}, 1 - \frac{1}{2}\frac{\lambda}{r_2} + \frac{1}{6}\frac{\lambda^2}{r_1r_2}\right).$$

Checking whether $||g(x) - g(y)|| < ||x - y||$ proves to be cumbersome, even for such a small network. Therefore we numerically analyzed the function $h(x, y) = ||g(x) - g(y)||(||x - y||)^{-1}$ which proved to be smaller than one for all values of x and y. As in the two node network, it is easy to show that for an instable network, either there is an obvious direct convergence to the rates (2/3,1/3,2/3) or convergence is proven by using parts of the approach for the stable case. We postulate that for any network a similar analysis will show that the algorithm constitutes a contraction, and thus converges.

A.2. Proof of Theorem 3

The formula for the rate $r_{i,\mathbf{1}}(n)$ of a node on position i in an n node network that is fully alive is given by

$$nr_{i,\mathbf{1}}(n) = \sum_{k=1}^{i-2} r_{i-k-1,\mathbf{1}}(n-k-1) + 1 + \sum_{k=i}^{n-2} r_{i,\mathbf{1}}(k) \tag{14}$$

as described in the paper. Due to symmetry of the network we also have that

$$r_{i,\mathbf{1}}(n) = r_{n-i+1,\mathbf{1}}(n) \quad i = 1, \ldots, n.$$

The rate of a node can never exceed one, but will be one if the node is the only alive node within its interference region, i.e. its neighbors are not alive. The minimal rate of a node is $1/n$ as with this probability it wins the contention over all other nodes.

In the following we omit the **1** denoting the fully alive network. To find an expression for $r_i(n)$, note that

$$nr_i(n) - (n-1)r_i(n-1) = r_i(n+2) + \sum_{k=1}^{i-2} [r_{i-k-1}(n-k-1) - r_{i-k-1}(n-k-2)]$$

and letting $c_i(n) = r_i(n) - r_i(n-1)$ this gives

$$c_i(n) = \frac{1}{n}\left[\sum_{k=1}^{i-2} c_k(n+k-i) - c_i(n-1)\right].$$

As $-1 \le c_i(n) \le 1$ for any value of i and n, we have that

$$c_i(n) \le \frac{1}{n}\left[(i-2) - c_i(n-1)\right] \le \frac{1}{n}(i-1)$$

so that for each i we have that $\lim_{n\to\infty} c_i(n) = 0$, proving that $r_i(n)$ converges for $n \to \infty$.

For $i = 1$ this leads to

$$c_1(n) = -\frac{1}{n}c_1(n-1)$$

which gives

$$c_1(n) = \frac{(-1)^{n-1}}{n!}, \quad r_1(n) = \sum_{i=1}^{n}\frac{(-1)^{i-1}}{i!}.$$

Similarly, we have that

$$c_2(n) = \frac{(-1)^{n}}{n!}, \quad r_2(n) = \sum_{i=1}^{n}\frac{(-1)^{i}}{i!}.$$

Taking the limit shows that

$$\lim_{n\to\infty} r_1(n) = 1 - \frac{1}{e}, \quad \lim_{n\to\infty} r_2(n) = \frac{1}{e}$$

Unfortunately, for larger values of i, no nice expressions are found for $c_i(n)$ or $r_i(n)$, but the limiting values can be calculated using the same approach. The results are presented in Table 1.

References

[1] Rao RR, Ephredimes A. On the stability of interacting queues in a multiple access system. IEEE Trans Inform Theory 1988;34(5):918–30.
[2] Jonckheere M, van der Mei RD, van der Weij W. Rate stability and output rates in queueing networks with shared resources. Technical Report, CWI, PNA-R0712; 2007.
[3] Borst SC, Jonckheere M, Leskelä L. Stability of parallel queueing systems with coupled service rates. Discrete Event Dyn Syst 2008;18(4):447–72.
[4] de Veciana G, Lee TJ, Konstantopoulos T. Stability and performance analysis of networks supporting elastic services. IEEE/ACM Trans Netw 2001;9(1):2–14.
[5] Denteneer D, Borst S, van de Ven P, Hiertz G. IEEE 802.11s and the Philosophers' problem. Stat Neerl 2008;62(3):283–98.
[6] Resing JAC, Örmeci L. A tandem queueing model with coupled processors. Oper Res Lett 2003;31:383–9.
[7] van Leeuwaarden JSH, Resing JAC. A tandem queue with coupled processors: computational issues. Queueing Syst 2006;51:29–52.
[8] Gupta P, Kumar PR. The capacity of wireless networks. IEEE Trans Inform Theory 2000;46(2):388–404.
[9] Gastpar M, Vetterli M. On the capacity of wireless networks: the relay case. In: Proceedings of INFOCOM 2002,. 2002. p. 1557–66.
[10] Ng PC, Liew SC. Offered load control in IEEE802.11 multi-hop ad-hoc networks. In: ICC 2002, vol. 2. 2002. p. 1074–9.
[11] Fujiwara T, Sekiya H, Namba K, Lu J, Yahagi T, Sakata S. Throughput an. on string multi-hop networks with multiple flow. In: NCSP '06. 2006. p. 417–20.
[12] Durvy M, Dousse O, Thiran P. Border effects, fairness, and phase transition in large wireless networks. In: Proceedings of INFOCOM 2008. 2008. p. 601–9.
[13] El-Khoury R, El-Azouzi R. Stability-throughput analysis in a multi-hop ad hoc networks with weighted fair queueing. In: 45th an. allerton conf. 2007. p. 1066–73.
[14] Kherani A, El Azouzi R, Altman E. Stability-throughput tradeoff and routing in multihop wireless ad-hoc networks. Comput Netw 2008;52(5):1365–89.
[15] Hekmat R, van Mieghem P. Interference in wireless multi-hop ad-hoc networks and its effect on network capacity. Wirel Netw 2004;10(4):389–99.
[16] Kelly FP. Reversibility and stochastic networks. Chichester: Wiley; 1979.
[17] Istratescu VI. Fixed point theory: an introduction. Mathematics and its application, vol. 7. Reidel; 1981.

Please cite this article in press as: Coenen T, et al. Bottlenecks and stability in networks with contending nodes. Int J Electron Commun (AEÜ) (2012), http://dx.doi.org/10.1016/j.aeue.2012.06.009