On the Nonlinear Dynamics of Network Flow Control Algorithms

Predrag R. Jelenković and Aurel A. Lazar

Department of Electrical Engineering
and
Center for Telecommunications Research
Columbia University, New York, NY 10027-6699

Abstract

The dynamical behavior of a class of synchronous nonlinear flow control algorithms is investigated. Under general monotonicity conditions on reaction functions we prove that the distributed algorithm cannot have non trivial period two equilibrias. When additional assumptions on the derivatives of reaction functions are imposed we prove the existence of a unique Nash Equilibrium (fixed) point and global convergence (with exponential rate) of the associated distributed algorithms. We also show that a relaxation of the monotonicity conditions may lead to chaotic behavior.

1 Introduction

Flow control algorithms regulate the input flow to the network, assigning appropriate loads to different user types. In practice these algorithms are distributed by the very nature of the network.

In this paper, network users (controllers) are modeled as noncooperative players that are competing for network resources according to their individual objectives. Users implement simple greedy algorithms: each user is optimizing its own objective function without the knowledge of the other players objective functions. We investigate global convergence of the greedy algorithms under general monotonicity conditions imposed on the users reaction functions. We extend the results of [8, 1, 16] that were obtained assuming linear reaction functions.

The rest of the paper is organized as follows. In section 2 we give the statement of the problem and a short overview of the existing results. In section 3 we extend these results to a general nonlinear setting. Finally, in section 4 we show that when monotonicity conditions are not satisfied the algorithms may exhibit chaotic behavior. The paper is concluded in section 5.

2 Problem Statement

We will restrict ourself to the analysis of an isolated bottleneck node in the network. First, assume n Poisson streams of packets with rates $\lambda_1, \ldots, \lambda_n$, that are serviced by a single server with exponential service rate μ. Each stream represents a single user's flow to the network. Further, each user is considered to be greedy, i.e., each user acts according to its own optimality objectives. The fundamental question is what is the resulting network dynamics under selfish user behavior.

Two commonly used objective criteria are the maximum throughput under time delay constraints $\max_{\mathbb{E}\tau \leq \mathrm{T}} \mathbb{E}\gamma$ [13], and the Power function $\mathbb{E}\gamma^\beta / \mathbb{E}\tau$, where parameter β may be used to achieve different trade off points between average throughput and expected time delay [1], [8].

In general, each user i can use an arbitrary objective function $P_i(\lambda_1, \ldots, \lambda_n)$. Our main objective is to understand the dynamics driven by an algorithm (game) in which an user optimizes noncooperatively its own objective function. In Game Theory [5] the equilibrium point of such a game is called a *Nash Equilibrium*, i.e., a point from which no player has an incentive to move from. More formally the Nash Equilibrium point is defined as follows.

Definition 1 *Let $\Lambda = \{(\lambda_1, \ldots, \lambda_n) : \sum_{i=1}^n \lambda_i < \mu\}$ be a set of admissible loads. Then an admissible load $(\lambda_1^*, \ldots, \lambda_n^*)$ is a Nash equilibrium iff*

$$
\begin{aligned}
&P_1(\lambda_1^*, \lambda_2^*, \ldots, \lambda_n^*) > P_1(\lambda_1, \lambda_2^*, \ldots, \lambda_n^*) && \forall \lambda_1 \in \Lambda_1 \\
&\vdots && \vdots \\
&P_i(\lambda_1^*, \ldots, \lambda_i^*, \ldots, \lambda_n^*) > P_i(\lambda_1^*, \ldots, \lambda_i, \ldots, \lambda_n^*) && \forall \lambda_i \in \Lambda_i \\
&\vdots && \vdots \\
&P_n(\lambda_1^*, \lambda_2^*, \ldots, \lambda_n^*) > P_n(\lambda_1^*, \lambda_2^*, \ldots, \lambda_n) && \forall \lambda_n \in \Lambda_n
\end{aligned}
$$

where $\Lambda_1, \ldots, \Lambda_n$ are the sets of all admissible loads for users $1, \ldots, n$, respectively, i. e., $\Lambda_i = \{\lambda_i : \lambda_i + \sum_{k \neq i} \lambda_k^* < \mu\}, 1 \leq i \leq n$.

There have been two types of algorithms considered in literature for reaching the Nash equilibrium: synchronous and asynchronous. In what follows we will focus on synchronous algorithms [1]. Assuming that each user can negotiate its service with the server during specific time slots, one user at a time, the algorithm is described as follows. At step k of the algo-

Published in: Proceedings of the Conference on Information Science and Systems, Princeton University, Princeton, New Jersey, March 20-22, 1996

rithm we have:

$$\lambda_1^{k+1} = R_1(\lambda_2^k, \ldots, \lambda_n^k)$$
$$\stackrel{def}{=} \arg\max_{\lambda_1 \in \Lambda_1} P_1(\lambda_1, \lambda_2^k, \ldots, \lambda_n^k)$$
$$\lambda_2^{k+1} = R_2(\lambda_1^{k+1}, \lambda_2, \lambda_3^k, \ldots, \lambda_n^k)$$
$$\stackrel{def}{=} \arg\max_{\lambda_2 \in \Lambda_2} P_2(\lambda_1^{k+1}, \lambda_2, \lambda_3^k, \ldots, \lambda_n^k)$$
$$\vdots$$
$$\lambda_n^{k+1} = R_n(\lambda_1^{k+1}, \lambda_2^{k+1}, \ldots, \lambda_{n-1}^{k+1})$$
$$\stackrel{def}{=} \arg\max_{\lambda_n \in \Lambda_n} P_n(\lambda_1^{k+1}, \lambda_2^{k+1}, \ldots, \lambda_{n-1}^{k+1}, \lambda_n)$$

with $(\lambda_2^0, \ldots, \lambda_n^0)$ being any admissible initial load and $R_1, \ldots, R_n$ the users reaction functions. We assume that once players choose their order of playing, this order is kept throughout.

From a dynamical systems point of view, notice that the reaction functions drive the dynamics of the game. The one step map $F = (F_1, \ldots, F_n) : \Lambda \rightarrow \Lambda$ of this dynamical system is determined by the reaction functions as follows:

$$\text{Let} \quad y_j = F_j(\lambda_1, \ldots, \lambda_n), \quad j = 1, \ldots, n,$$

then, F is determined using the following recursive formula:

$$y_1 = R_1(\lambda_1, \ldots, \lambda_n)$$
$$y_j = R_j(y_1, \ldots, y_{j-1}, \lambda_j, \ldots, \lambda_n), \qquad j = 2, \ldots, n.$$

Thus, the evolution of this algorithm is equivalent to the dynamics described with:

$$\lambda^{k+1} = F(\lambda^k), \tag{1}$$

where $\lambda^k = (\lambda_1^k, \ldots, \lambda_n^k)$. (It is easy to show that F maps $\Lambda \rightarrow \Lambda$ and therefore iterations of all order are well defined.)

Convergence to the Nash Equilibrium has being investigated in the literature both for the maximum throughput and the Power function objective criterion in [1]. These results have been further extended in [16]. In all these results reaction functions were linear. For example the reaction function for the Power function optimization objective is given by

$$R_i(\lambda_1, \ldots, \lambda_n) = \frac{\beta_i}{1+\beta_i}(\mu - \lambda_{-i}), \tag{2}$$

where $\lambda_{-i} = \sum_{k \neq i} \lambda_k$.

3 Results

In this section we extend the results from [1, 16] to the nonlinear case. Instead of first trying to come up with specific objective functions, we postulate a set of reaction functions, that completely determine the dynamics of the game. We will choose users reaction functions based on the basic underlying properties of these functions that came to be as in the tractable Poisson case and maximum throughput/Power objective criterion. From the analysis conducted in [1] (see equation (2)), we observe that reaction functions R_i for the maximum throughput/Power objectives are:

a) functions of the total load of all other users (λ_{-i}) except user i.

b) linear in λ_{-i}.

c) monotonically decreasing in λ_{-i}.

Our objective is to relax condition b), and therefore we define:

Definition 2 *Let $\mathcal{M}_i$ be the set of all reaction functions for user $i \in [1, n]$, such that, any $R_i \in \mathcal{M}_i$ satisfies conditions a), c). We call $\mathcal{M}_i$ a set of admissible reaction functions for user i; the game in which each user is using an admissible reaction function is called a fair game.*

The monotonicity assumption is also natural in other network control algorithms that use a noncooperative game theoretic setting. For example, in routing algorithms the reaction functions are also monotonically decreasing in the total load of all the other users (see [12]).

If the total demand of all users except user's i is increasing, the demand of user i should be decreasing. This means that user's i reaction is to accommodate the load of the other users, i.e., fairness property. We will see that monotonicity and fairness are further closely connected to the convergence (stability) of the algorithm. Let us first state the following simple result for the case of the two user game.

Proposition 1 *In a single server communication system consisting of two users with reaction functions $R_i \in \mathcal{M}_i, i = 1, 2$, competing for a server with capacity μ, the successive iterates of the flow converge monotonically to a fixed point in Λ for any initial condition $\Lambda^0 = (\lambda_1^0, \lambda_2^0) \in \Lambda$.*

Remark: Note that there may be more than one NEP; multiple equilibria have been found for the maximum throughput criterion in [11]. To ensure the uniqueness of the NEP more restrictive conditions will be needed.

Proof: The dynamics of this game is completely defined by the function $f = R_1 \circ R_2$. Then, by the chain rule $f' = R_1'(R_2)R_2' > 0$, since both R_1, R_2 are monotonically decreasing, and by Lemma 3.14. pp. 82 [10], f displays monotone dynamics. Since f is bounded $(f : [0, \mu] \rightarrow [0, \mu])$, each orbit of f is convergent. This proves the proposition. $\diamond$

We see that convergence follows from monotonicity (and boundedness of the load space). Unfortunately, for the distributed algorithm with more than two users the underlying monotonicity property is lost; this can be seen in Figure 3, where we present the loads of three players during the first 10 iterations. The reaction functions of all users are the same and given by $R(x) = \pi - 2\arctan(x - 10), x \geq 0$ (assume that the capacity of the server is large enough to accomodate all possible reactions). Then, it is clear (from Figure 3) that the loads of two of the users are not monotonic. However, the system is still relatively constrained, and we can prove the following result. In what follows the vectors λ^1 and λ^2 are said to be ordered $\lambda^1 <> \lambda^2$ if $\lambda_i^1 \leq\geq \lambda_i^2, 1 \leq i \leq N$ with inequality being strict for at least one i.

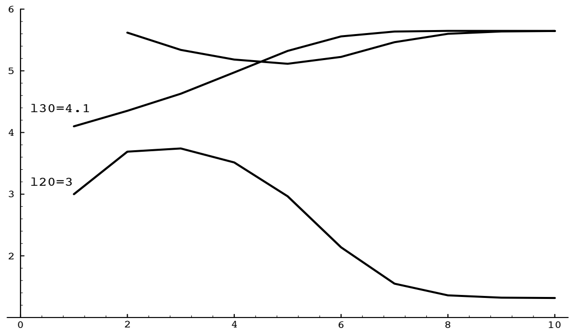

Figure 1: Example of non monotonic sample paths in the game of three users.

Theorem 1 *In a fair (monotone) flow control algorithm, the following properties hold:*

a) it is not possible to have

$$\lambda^{k+1} > \lambda^k, \quad or \ \lambda^{k+1} < \lambda^k,$$

for any $k \geq 0$.

b) the system cannot have nontrivial period 2 equilibrium, that is, $\lambda^* = F^2(\lambda^*)$, *and* $\lambda^* \neq F(\lambda^*)$.

Proof: Statement a) is almost immediate. If vectors $\lambda_{-n}^k \stackrel{def}{=} (\lambda_1^k, \ldots, \lambda_{n-1}^k)$ and $\lambda_{-n}^{k+1} \stackrel{def}{=} (\lambda_1^{k+1}, \ldots, \lambda_{n-1}^{k+1})$ are not ordered the statement immediately follows. If they are ordered then, either $\lambda_{-n}^k = \lambda_{-n}^{k+1}$, $\lambda_{-n}^k < \lambda_{-n}^{k+1}$, or $\lambda_{-n}^k > \lambda_{-n}^{k+1}$; the equality implies that $\lambda^k = \lambda^{k+1}$, and the inequalities $\lambda_{-n}^k >< \lambda_{-n}^{k+1}$ imply the reverse inequalities for the last coordinate, i.e., $\lambda_n^k <> \lambda_n^{k+1}$. This proves a).

For b) we will assume that there is a period 2 equilibrium, i.e., that there exist $\lambda^1 \neq \lambda^2$, such that $\lambda^2 = F(\lambda^1)$ and $\lambda^1 = F^2(\lambda^1)$. Let us denote the distance vector as $\delta = \lambda^1 - \lambda^2 \neq 0$. Then, we have the following set of equations

$$\delta_i = \lambda_i^1 - \lambda_i^2 \Leftrightarrow \lambda_{-i}^1 - \lambda_{-i}^2 = -w_i \delta_i,$$

for some $w_i \geq 0$, and we take the convention that $w_i = 0$ if $\delta_i = 0$, $1 \leq i \leq n$. Then, from the system of equations above, δ satisfies the following homogenous system of equations

$$W\delta = 0, \tag{3}$$

where

$$W = \begin{pmatrix} w_1 & -1 & -1 & \ldots & -1 \\ 1 & w_2 & -1 & \ldots & -1 \\ 1 & 1 & w_3 & \ldots & -1 \\ \vdots & & & & \vdots \\ 1 & 1 & 1 & \cdots & w_n \end{pmatrix}. \tag{4}$$

For this system to have a nonzero solution it is necessary that $\det W = 0$. However, it is not very difficult to prove (say by induction) that

$$\det W = \begin{cases} \sum_{i=0}^{(n-1)/2} \sum_{k_1 \neq \cdots \neq k_{2i+1}} w_1 \cdots w_{k_{2i+1}} \\ \qquad\qquad \text{if n is odd,} \\ 1 + \sum_{i=1}^{n/2} \sum_{k_1 \neq \cdots \neq k_{2i+1}} w_1 \cdots w_{k_{2i}} \\ \qquad\qquad \text{if n is even.} \end{cases}$$

Therefore, $\delta \neq 0$ implies $\det W > 0$ which contradicts the fact that δ is a solution to (3). This provides the proof of b). $\diamond$

Result a) (in the preceding theorem) says that in one round of control it is not possible for every user to increase its load, i.e., if some of the users increase their load, others have to decrease it. For that reason we call this type of algorithm *competitive.*

Theorem 1 is a promising result in the direction of more closely characterizing the global dynamics of the fair competitive algorithms. It is an open problem if there exists period three (or higher) equilibrias. For example for period three equilibrias the proof of the preceding theorem doesn't go through. On the other hand, the dificulty of finding a counterexample is in having to work in more than three dimensions.

Some intuition from well studied *competitive* fluid models [15, 7] suggests that we should expect complex dynamics in higher dimensions for our discrete time competitive schemas. However, how complex, for example, whether our system can display chaotic behavior remains an open question.

It is known that *cooperative* fluid systems are very stable [6], [7]. The main result is quasi-convergence almost everywhere, i.e., for almost any initial condition the system converges to the set of equilibrias [6]. Similarly, for our discrete time algorithms it is not very difficult to see how cooperation can stabilize the system. Let's assume that, controllers are devided into two groups, such that one group is competing with the other, i.e., the joint reaction of one group is monotonicaly decreasing in the load of the other group. Then, the algorithm that devides the bandwith between the two groups is stable (Proposition 1) and therefore, each subset of users can further compete for the bandwith, i.e., each subgroup can be subdivided into cooperative groups, and so on. However, cooperation is very difficult to implement in a distributive system. For that reason we don't pursue this matter further.

When additional assumptions on the decreasing rates of the reaction functions are imposed, the algorithm becomes globaly stable.

Definition 3 *For each* $i \in [1, n]$, $\mathcal{M}_i^{e,q} \subset \mathcal{M}_i$ *is a set of* (e, q) *admissible r.f.s of user* i, *such that* $0 < e \leq |R_i'(\lambda_{-i})| \leq q < 1$ $(R_i' = \frac{dR_i}{d\lambda_{-i}})$.

The following theorem is our main result.

Theorem 2 *Consider* n *users competing for the server of capacity* μ. *Each user* i *has a r.f.* $R_i \in$

$\mathcal{M}_i^{e,q}$. *Then F has a unique fixed point (Nash Equilibrium point), and for every initial condition* $\lambda^0 \in \Lambda$, $\{F^n(\lambda^0)\}_{n\in\mathbb{N}}$ *converges exponentionally fast to that point, as* $k \to \infty$.

Proof: Given in Appendinx. ◇

By droping the monotonicity conditions, i.e., allowing users to be unfair, we present several examples in the following section for which the dynamics of the algorithms can exhibit arbitrary, including chaotic, behavior [3].

4 Game of Two Users with Dishonest Reaction Functions

We have seen in the last section that in the case of monotone (stable) decreasing reaction functions, in the system with two users monotone dynamics (Proposition 1) arise. What can happen if one of the users behaves improperly, so that his reaction function is not monotonically decreasing? Then, literally anything can happen.

To see this let us suppose that one user has reaction function $R_1(\lambda_2) = \lambda_2$ if $\lambda_2 < 1$, and zero otherwise, and the other has $R_2(\lambda_1) = c\lambda_1(1-\lambda_1), 0 < c \le 4$ in the interval $[0,1]$ and zero otherwise. Then, chaotic behavior of the system results, since $f = R_1 \circ R_2$ is the logistic map between zero and one, which is well known in the literature of dynamical systems (first 158 pages in [3] are devoted to this map). Instead of repeating the whole story about this map, we will only mention a few characteristics: it exhibits chaotic, periodic and almost all other phenomena that occur in one dimensional dynamics.

Is this maping the only one which can exibit chaotic behavior? Of course not, even piecewise linear maps can be chaotic. Assume that the first user has the same reaction function as above and the second user having the Tent map as reaction function:

$$R_2(\lambda) = \begin{cases} 2\lambda & 0 \le \lambda \le 0.5 \\ 2(1-\lambda) & 0.5 \le \lambda \le 1. \end{cases}$$

Then, a composition of this two reaction functions is again the Tent map. It is somewhat surprising that this nice looking map, is chaotic, and moreover, topologically conjugate with the quadratic map $f = 4\lambda(1-\lambda)$. In Figure 4 the first five hundred iterations of this map are given, and the resulting chaotic behavior is evident.

One can argue about these examples as being artificial, because they exhibit the anomalous property that a user is sending no traffic into the idle network, and sending some positive flow when the network gets congested. Now, we will see how in a more realistic situation the system can exibit a periodic global behavior. Consider two users with maximum throughput minimum time delay objectives ($\max_{\mathbb{E}T_i \le \mathrm{T}_i} \lambda_i, i = 1,2$). Then, the reaction functions are linearly decreasing, and if $T_1 < T_2$ there is one globaly attracting point in this system which corespond to the maximum flow of the user with the more relaxed time delay bound and zero flow for the other user. This is obviously unfavorable to the user with more restrictive bounds, and one way for him/her to improve throughput is to

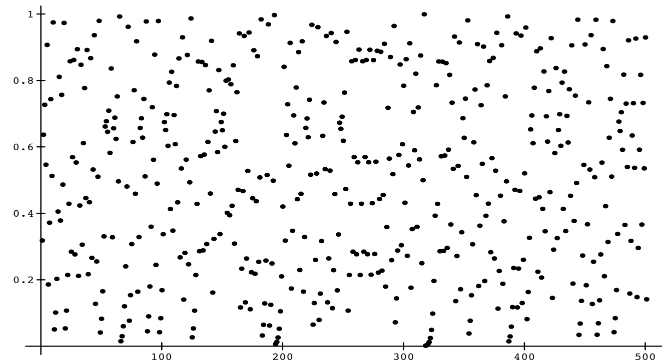

Figure 2: First five hundred iterations of the Tent map with initial condition $x_0 = 1/\pi$.

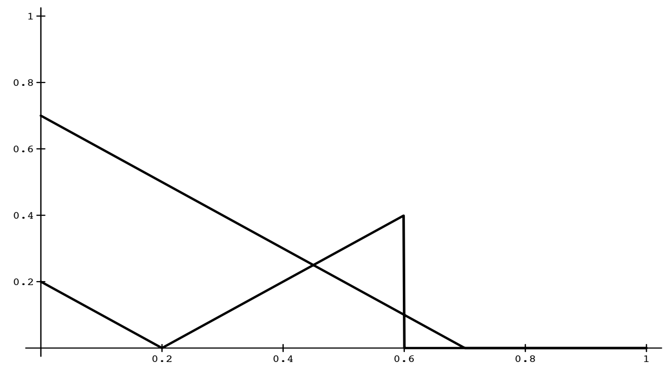

Figure 3: An example of reaction functions where one user is honest (linearly decreasingreaction function) and the other is perturbing the system.

start sending some non-valuable flow to the network in order to perturb the system and make the other user reduce his/her load. Examples of such reaction functions are presented in Figure 4.

The dynamics of this algorithm can be easily evaluated by analyzing the plot of $R_1 \circ R_2$. As can be seen from Figure 4, the system has three fixed points. The only stable fixed point is $x = 0.7$. Further, for initial values inside $[0.1, 0.3]$ and $[0.6, 1]$ the system converges globaly to the fixed point $x = 0.7$ and otherwise it is periodical with period two. Thus, inside the intervals $[0, 0.1]$ and $[0.3, 0.6]$ the second user gains from the system by perturbing it. That is, if a user with high objectives (second user) sends a big nonvaluable load into the network that makes the user with lower objectives to reduce his/her load, then the second user reacts by sending a small valuable load making the first user to increase his/her load, and so on.

Therefore, even with two users very complex dynamics can arise and general results on global convergence appear difficult to obtain on this level of generality.

5 Conclusion

The dynamic behavior of a class of synchronous nonlinear flow control algorithms was investigated. Under general monotonicity conditions on reaction functions the associated distributed algorithm cannot have non trivial period two equilibrias. When addi-

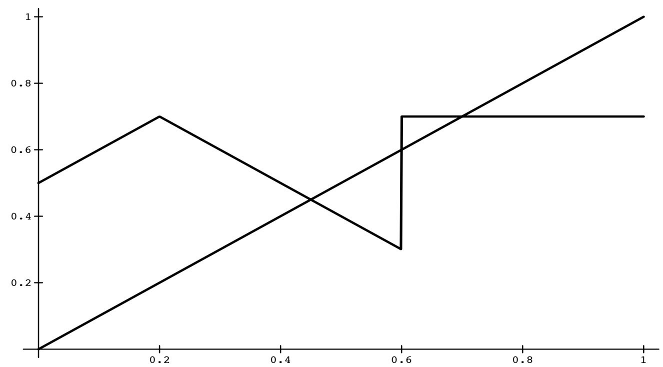

Figure 4: Plot of the composition of the reaction functions given in the figure above.

tional assumptions on the derivatives of reaction functions are imposed the existence of a unique Nash Equilibrium (fixed) point and global convergence (with exponential rate) of the associated distributed algorithms was shown.

We addressed several open problems for general monotone competitive algorithms, and discussed their connection to well studied competitive and cooperative fluid systems. We showed how, by adding cooperation, a competitive algorithm may be stabilized. Connections between fairness, monotonicity, competition and stability are demonstrated. We also showed that a relaxation of the monotonicity condition in Theorems 1, 2 may lead to chaotic dynamics.

6 Appendix

Let us now state a useful algebraic result

$$\det\begin{pmatrix} a_1 & x & x & \dots & x \\ x & a_2 & x & \dots & x \\ x & x & a_3 & \dots & x \\ \vdots & & & & \vdots \\ x & x & x & \dots & a_n \end{pmatrix} \tag{5}$$
$$= x(a_1-x)(a_2-x)\cdots(a_n-x) \times\left(\tfrac{1}{x}+\tfrac{1}{a_1-x}+\tfrac{1}{a_2-x}+\cdots+\tfrac{1}{a_n-x}\right).$$

(**Computational hint:** Subtract the first row from all the others, then take out common factors $(x-a_i)$ and add all columns to the first one. To finish the derivation, develop the determinat using first column.)

Proof of Theorem 2: The idea is to prove that F is a contraction mapping, and then from Banach's Contraction Theorem (see [4], pp. 156) the conclusion of the theorem will follow.

For fixed $\lambda = (\lambda_1, \dots, \lambda_n) \in \Lambda$, denote with $a_i^\lambda = -\frac{dR_i(\lambda_{-i})}{d\lambda_{-i}}, 1 \le i \le N$ (recall $\lambda_{-i} = \sum_{k=1}^N \lambda_k - \lambda_i$). Then, the first derivative of F at λ (Jacobian) is given by

$$F_\lambda^{'} = \begin{pmatrix} \frac{1}{a_1^\lambda} & 0 & 0 & \cdots & 0 \\ 1 & \frac{1}{a_2^\lambda} & 0 & \cdots & 0 \\ 1 & 1 & \frac{1}{a_3^\lambda} & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & 1 & \cdots & 1 & \frac{1}{a_N^\lambda} \end{pmatrix}^{-1} \times \begin{pmatrix} 0 & -1 & -1 & \cdots & -1 \\ 0 & 0 & -1 & \cdots & -1 \\ 0 & 0 & 0 & \cdots & -1 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & -1 \end{pmatrix}.$$

(Derivation of this equation is not very difficult, and for that reason we don't go into details.) Then, it is not hard to see that $F_\lambda^{'}$ is exactly the Gauss-Seidel matrix for solving the system of linear equations $Ax = b$, where

$$A^\lambda = \begin{pmatrix} \frac{1}{a_1^\lambda} & 1 & 1 & \dots & 1 \\ 1 & \frac{1}{a_2^\lambda} & 1 & \dots & 1 \\ 1 & 1 & \frac{1}{a_3^\lambda} & \cdots & 1 \\ \vdots & & & & \vdots \\ 1 & 1 & 1 & \cdots & \frac{1}{a_N^\lambda} \end{pmatrix} \tag{6}$$

Let D, B, C be diagonal, left lower, and right upper triangular matricies, respectively, such that $A^\lambda = B + D + C$; further, let $K \stackrel{def}{=} (F_\lambda^{'})^{-1}BD^{-1}C$. Then, by applying Theorem 16 pp. 143 [4], we have

$$\|F_\lambda^{'}\|_A = \frac{\mu}{1+\mu} < 1, \tag{7}$$

for all $\lambda \in \lambda$; μ is the maximum eigenvalue of K^λ, and $\|\cdot\|_A$ is the norm on Λ induced by matrix A^λ (for more details see [4], pp. 143).

To finish the proof we will use Lipschitz's criterion ([4], pp. 157) for contraction mappings, i.e., F is a contraction if there exists a norm such that $\sup_{\lambda\in\Lambda} \|F_\lambda^{'}\| < 1$. Now, we will proceed to find an appropriate norm for which this will hold. Let

$$Q = \begin{pmatrix} \frac{1}{q} & 1 & 1 & \dots & 1 \\ 1 & \frac{1}{q} & 1 & \dots & 1 \\ 1 & 1 & \frac{1}{q} & \dots & 1 \\ \vdots & & & & \vdots \\ 1 & 1 & 1 & \dots & \frac{1}{q} \end{pmatrix}. \tag{8}$$

Note also that matrices A^λ and Q induce (since they are symmeric and positive definite) norms on Λ

$$\|x\|_{A^\lambda} = (A^\lambda x, x)_2, \;\; \|x\|_Q = (Qx, x)_2 \quad x \in \Lambda \tag{9}$$

where $(\cdot,\cdot)_2$ is usual quadratic norm on Λ.

Also, observe that for fixed $\lambda \in \Lambda$ and any $y \in \Lambda$

$$\begin{aligned} ||y||_Q &= (\frac{1}{q}-1)\sum_{i=1}^{N} y_i^2 + (\sum_{i=1}^{N} y_i)^2 && (10) \\ &\leq \sum_{i=1}^{N}(\frac{1}{a_i^\lambda}-1)y_i^2 + (\sum_{i=1}^{N} y_i)^2 && (11) \\ &= ||y||_{A^\lambda} && (12) \end{aligned}$$

This further implies

$$||F_\lambda'||_Q \leq ||F_\lambda'||_{A^\lambda} = \frac{\mu_\lambda}{1+\mu_\lambda} < 1 \qquad (13)$$

Now, in order to preserve strict inequality after taking supremum over all $\lambda \in L$ we have to prove that μ_λ is bounded by a constant for all $\lambda \in L$. Then, it will follow that $||F_\lambda'||_Q \leq \frac{C}{C+1} < 1$, for all $\lambda \in L$ (and, as we already mentioned, by Lipschitz's criterion it will follow that F is a contraction mapping). Therefore, to finish the proof we have to show that μ_λ is uniformly bouded. To show that, note

$$\begin{aligned} \mu_\lambda &\leq ||K||_\infty \leq ||(F_\lambda')^{-1}||_\infty ||B||_\infty ||D^{-1}||_\infty ||C||_\infty \\ &\leq ||(F_\lambda')^{-1}||_\infty n^2 q, \end{aligned}$$

where the first inequality follows from the fact that the maximal eigenvalue of any matrix is smaller than any norm of that matrix; second inequality follows from the well known result that the norm of the product of matrices is smaller than the product of the norms; and the last inequality follows from the rough upper bound on $||B||_\infty$,$||D^{-1}||_\infty$, and $||C||_\infty$. Therefore, it is left to upperbound $||(F_\lambda')^{-1}||$ to show uniform boundedness. Let $[b_{ij}]_1^n = (F_\lambda')^{-1}$, then, according to Cramers rule, $|b_{ij}| = \frac{|d_{ij}|_\lambda}{|d|_\lambda}$, where d and d_{ij} are determinant and ij-th cofactor of the matrix F_λ'. Utilizing the algebraic identity from the previous page we calculate

$$\begin{aligned} |d_\lambda| &= (\frac{1}{a_1}-1)\cdots(\frac{1}{a_n}-1) && (14) \\ &\quad \times(1+\frac{a_1}{1-a_1}+\cdots+\frac{a_n}{1-a_n}) \\ &\geq (\frac{1}{q}-1)^n. \end{aligned}$$

Similarlly, from $a_i \geq \epsilon$ we obtain an upper bound

$$|d_{ij}| \leq n!\frac{1}{\epsilon^n}. \qquad (15)$$

From (14, 15) it follows

$$|b_{ij}| \leq n!(\frac{1-q}{q\epsilon})^n = C, \qquad (16)$$

which implies

$$||(F_\lambda')^{-1}||_\infty \leq nC. \qquad (17)$$

This proves the theorem.

◇

References

[1] A. D. Bovopolous and A. A. Lazar. Decentralized algorithms for optimal flow control. In *Proceedings of the 25th Allerton Conference on Communications Control and Computing*, University of Illinois at Urbana-Champaign, Oct 1987.

[2] Bertsekas D. and Gallager R. *Data Networks.* Prentice-Hall, 1987.

[3] R. L. Deavaney. *An Introduction to Chaotic Dynamical Systems.* Addison-Wesley, 1989.

[4] Stummel F. and Hainer K. *Introduction to Numerical Analysis.* Scottish Academic Press, 1980.

[5] D. Fudenberg and J. Tirole. *Game Theory.* The MIT Press, 1991.

[6] M. Hirsch. Systems of differential equations which are competitive or cooperative ii: convergence almost everywhere. *SIAM J. Math. Anal.*, 16:423–439, 1985.

[7] M. Hirsch. Systems of differential equations which are competitive or cooperative iii. competing species. *Nonlinearity*, 1:51–71, 1988.

[8] M. T. Hsiao and A. A. Lazar. Optimal decentralized flow control of markovian queueing networks with multiple controllers. *Perform. Eval.*, 13(3):181–204, 1991.

[9] Guckenheimer J. and P. Holmes. *Nonlinear Oscilations, Dynamical Systems, and Bifurcations of Vector Fields.* Springer-Verlag, 1983.

[10] Hale J. K. and Kocak H. *Dynamics and Bifurcations.* Springer-Verlag, 1992.

[11] Y. A. Korilis and A. A. Lazar. On the existence of equilibria in noncooperative optimal flow control. *Journal of the Association for Computing Machinery*, 42(3):584, May 1995.

[12] Y. A. Korilis, A. A. Lazar, and A. Orda. Architecting noncooperative networks. *IEEE Journal on Selected Areas in Communications*, 13(7):1241–1251, sep 1995.

[13] Hsiao M. T. and Lazar A. A. A game theoretic approach to optimal decentralized flow control of markovian queueing networks with multiple controllers. In *Proceedings of Performance87*, Brussels,Belgium, Dec 1987.

[14] Hirsch M. W. and S. Smale. *Differential Equations, Dynamical Systems, and Linear Algebra.* Academic Press, Inc., 1974.

[15] S. Smale. On the differential equations of species in competition. *J. Math. Biol.*, 3:5–7, 1976.

[16] Zhang Z. and Douligeris C. Convergence of synchronous greedy algorithms in a multiclass telecommunication environment. *IEEE Transactions on Commmunications*, 40(8):1277–1281, 1992.