Convergence of Synchronous and Asynchronous Greedy Algorithms in a Multiclass Telecommunications Environment

Zhensheng Zhang and Christos Douligeris

Abstract—In this letter, the Nash equilibrium point for optimum flow control in noncooperative multiclass environment is studied. Convergence properties of synchronous and asynchronous greedy algorithms are investigated in the case where several users compete for the resources of a single queue using power as their performance criterion. The main contributions of this letter consist of the proof of the convergence of a synchronous greedy algorithm for the \( n \) users case and the obtaining of the necessary and sufficient conditions for the convergence of asynchronous greedy algorithms. Another very important contribution of this letter is the introduction to the literature and the extension of a not very widely known theorem for the convergence of Gauss–Seidel algorithms in the linear systems theory.

I. INTRODUCTION

Flow control is necessary in a telecommunications network environment to ensure that the limited existing resources are used in a prudent and efficient way. Thus, unnecessarily long delays or deadlocks are avoided. Techniques to achieve optimum flow control involve the optimization of certain performance measures, like the maximization of the throughput, the minimization of the average delay, the minimization of blocked calls or several combinations of the above.

A multiclass environment arises as several user-classes or different origin-destination pairs have different performance objectives. How can an optimum flow control strategy be defined in such an environment where the performance objectives are often noncompatible and usually conflicting? Game theoretic approaches to the flow control problem have been studied by various authors, see for example, [13], [2], [8], and the references therein. This approach allows the consideration of the individual objectives as well as the performance of the network as a whole. It also makes possible the study of fairness in such an environment as well as the analysis of the influence of different information structures on the definition of an optimum point.

Nash equilibria (NE) [17] will be the subject of this letter. The Nash equilibrium point is defined as the point where no user would unilaterally deviate, because such a deviation would result in a worsening of its performance objective. NE have been studied rather extensively in the literature of flow control in packet switched networks, where users have only local information [5], [12], [2], [8]. In [15] it was noted that a decentralized algorithm according to which users optimized their individual performance measures—power in this particular case—was converging to an equilibrium point. Due to its structure the algorithm was termed greedy. It was also noticed that there were sets of throughputs that provided better performance than the one achieved by the greedy algorithm. Thus power as a performance measure was termed nondecentralizable [14]. Moreover, it was shown that asynchronous implementations of the greedy algorithm were either nonconvergent or oscillating.

In [8] and [10], it was shown that the point the decentralized algorithm was converging was a Nash equilibrium point. The Nash equilibrium was also calculated by iteratively solving a linear set of equations. Convergence of the greedy algorithm to the Nash equilibrium was shown for the two classes case. In [2], using the observation that the greedy algorithm corresponds to the Gauss–Seidel technique for solving a linear system of equations, it was shown that an asynchronous version of the algorithm does not converge when all the weighting factors were greater than or equal to one. In [5], the noncooperative Nash equilibrium was studied and a convergence proof was provided for the two users case only under several convexity conditions that are not satisfied in the case of weighting factors greater than one. In [9], [3], and [4] several properties on the convergence of synchronous and asynchronous versions of the algorithm were reported. A general convergence proof was not provided—even though numerical results pointed to that direction even for the more general case of a BCMP network.

In this letter we study the problem analyzed in the above mentioned works and show the convergence of the synchronous greedy algorithm to the Nash equilibrium for the general case of \( n \) noncooperative users competing for the resources of a single queue. Our proof takes a different approach than the ones presented in the works reported above, in that it does not study the Gauss–Seidel or the Jacobi matrix directly but after a transformation is made. We also find the necessary and sufficient conditions for convergence of asynchronous versions of the greedy algorithm in the general case.

Having proven the convergence of the synchronous greedy algorithm and having found the necessary and sufficient con-
ditions for the convergence of the asynchronous in the general case of $n$ users are the main contributions of our letter. Another very important contribution of this letter is Lemma 1 and its proof, i.e., the introduction to the literature and the extension of a not very widely known theorem for the convergence of Gauss-Seidel algorithms in the linear systems theory.

The letter is organized as follows. In Section II we present the model, the performance criterion and the definition of the Nash equilibrium and the greedy algorithm. In Section III we give the proof of the convergence of the greedy algorithm. In Section IV we analyze the convergence of the asynchronous algorithm. Section V concludes this letter. The proof of Lemma 1 is relegated to the Appendix.

II. STATEMENT OF THE PROBLEM

A. The Model

Consider $n$ Poisson streams of packets with arrival rates $\lambda_1, \ldots, \lambda_n$, that are serviced by a single exponential server with service rate $\mu$.

Each class of packets has as a performance objective the maximization of its power [11]—defined as the weighted ratio of the throughput over the average delay that the traffic experiences in that queue. To achieve that, each class adjusts its throughput, i.e., by selectively discarding packets. Let $\gamma_i$ be the resulting throughput of class $i$, $D_i(\gamma_1, \ldots, \gamma_n)$ the average delay and $\beta_i (\beta_i > 0)$ the weighting factor. Then $P_i = \gamma_i^n / D_i(\gamma_1, \ldots, \gamma_n)$ is the performance objective of class $i$.

A $n$-tuple $\gamma = (\gamma_1, \gamma_2, \ldots, \gamma_n)$ of nonnegative real numbers defines an admissible set of throughputs iff $\sum_{i=1}^{n} \gamma_i < \mu$.

B. The Nash Equilibrium

When no cooperation is allowed and the users are symmetric in the information they acquire, each one would try to reach a point so that its performance is optimal in the sense that any unilateral deviation from that point would render it worse off. This point is called the Nash equilibrium point. A more technical definition of the Nash equilibrium is given below.

Definition 1) Let $P_i(\gamma_1, \gamma_2, \ldots, \gamma_{i-1}, \gamma_i, \gamma_{i+1}, \ldots, \gamma_n)$ be the power of user $i$, $i = 1, 2, \ldots, n$. Then an admissible point $(\gamma_1, \gamma_2, \ldots, \gamma_n)$ is a Nash equilibrium solution iff

\[ P_i(\gamma_1, \gamma_2, \ldots, \gamma_n) > P_i(\gamma_1', \gamma_2, \ldots, \gamma_n) \quad \forall \gamma_i \in \Gamma_i \]

\[ \vdots \]

\[ P_i(\gamma_1, \gamma_2, \ldots, \gamma_n) > P_i(\gamma_1, \gamma_2, \ldots, \gamma_n') \quad \forall \gamma_i \in \Gamma_i \]

\[ \vdots \]

\[ P_n(\gamma_1, \gamma_2, \ldots, \gamma_n) > P_n(\gamma_1, \gamma_2, \ldots, \gamma_n) \quad \forall \gamma_n \in \Gamma_n \]

where $\Gamma_1, \Gamma_2, \ldots, \Gamma_n$ are the sets of all admissible throughputs for users 1, 2, $\ldots$, $n$, respectively.

It was shown in [8] that the Nash equilibrium point for the model presented in the previous sections is given by

\[ \gamma_i = \frac{\beta_i \mu}{1 + \sum_{k=1}^{n} \beta_k} \quad \text{for } i = 1, 2, \ldots, n. \]

C. The Synchronous Greedy Algorithm

In [14] an algorithm was presented that used only information locally available to each user. The algorithm was called greedy because each user was trying to optimize its performance without regarding the performance of the other users. This algorithm falls into a general category of algorithms that follow an iterative procedure, at each step of which all users update their control in an optimum way [9], [15], [16].

There is a particular order according to which users make the updates. The equilibrium point—if it exists—is independent of that particular order.

For the particular case of the power function this algorithm is the synchronous greedy algorithm and can be described as follows:

At step $k$ of the algorithm

\[ \gamma_1^{k+1} = \arg \max_{\gamma_1 \in \Gamma_1} P_1(\gamma_1, \gamma_2^k, \ldots, \gamma_n^k) \]

\[ \gamma_2^{k+1} = \arg \max_{\gamma_2 \in \Gamma_2} P_2(\gamma_1^k, \gamma_2, \gamma_3^k, \ldots, \gamma_n^k) \]

\[ \vdots \]

\[ \gamma_n^{k+1} = \arg \max_{\gamma_n \in \Gamma_n} P_n(\gamma_1^{k+1}, \gamma_2^{k+1}, \ldots, \gamma_{n-1}^{k+1}, \gamma_n) \]

with $(\gamma_1^0, \gamma_2^0, \ldots, \gamma_n^0)$ admissible but arbitrary.

The given order is not unique. Any order would suffice, provided that is kept throughout. In order to study the convergence properties of the algorithm the theorems presented in [16] cannot be used since power is not a concave function. But it can be easily seen that the Nash equilibrium point corresponds to the point $\gamma^*$, where

\[ \frac{\partial P_i}{\partial \gamma_i} \bigg|_{\gamma = \gamma^*} = 0 \]

which in the case of our model turns out to be the solution of the linear system of equations $Ax = b$, where

\[ A = \begin{pmatrix} 1 & \beta_1 & \ldots & \beta_1 \\ \beta_2 & \beta_2 + 1 & \ldots & \beta_2 \\ \vdots & \vdots & \ddots & \vdots \\ \beta_n & \beta_n + 1 & \ldots & 1 \end{pmatrix} \]

and

\[ b = \begin{pmatrix} \frac{\beta_1 \mu}{\beta_1 + 1} \\ \frac{\beta_2 \mu}{\beta_2 + 1} \\ \vdots \\ \frac{\beta_n \mu}{\beta_n + 1} \end{pmatrix} \]

III. CONVERGENCE OF THE SYNCHRONOUS GREEDY ALGORITHM

The synchronous greedy algorithm is then precisely the Gauss-Seidel method to iteratively solve the linear system.

In the following, the two names, synchronous greedy and Gauss-Seidel, will be used interchangeably. The main question that arises is under what conditions does the algorithm converge to the Nash equilibrium?

Previous attempts to study the convergence of the algorithm had limited success, due to the lack of symmetry and positive definiteness of matrix $A$. The use of techniques that required the sign of the roots of certain polynomials proved to be very
cumbersome and impractical for cases with more than three users.

In [9], it was shown that the algorithm converges to a Nash equilibrium for all initial conditions if all the $\beta_i$'s are equal, for all $n \geq 1$. The convergence of the general $\beta_i$'s was only proven for the case $n = 2$ and $n = 3$. In the following, we prove that the algorithm does converge for all $\beta_i$'s for all $n \geq 1$. For simplicity, we define the diagonal matrix $D$ as

$$D = \text{diag} \left( \frac{1 + \beta_1}{\beta_1}, \ldots, \frac{1 + \beta_n}{\beta_n} \right)$$

and let

$$\bar{A} = DA.$$ 

Note that matrix $\bar{A}$ is now symmetric and positive definite and all the theorems that study the convergence of the Gauss-Seidel technique can now be applied. Theorem 1 uses this property to provide the convergence proof of the greedy algorithm.

**Theorem 1:** The greedy algorithm converges to a Nash equilibrium for all initial conditions, for all $n \geq 1$.

**Proof:** Let $M_G$ ($\bar{M}_G$, respectively) be the lower triangle matrix of $A$ (\A, respectively) and $N_G = M_G - A$ ($\bar{N}_G = \bar{M}_G - \bar{A}$, respectively). Define the Gauss-Seidel matrix $Q_G = M_G^{-1}N_G$ and $\bar{Q}_G = \bar{M}_G^{-1}\bar{N}_G$. Then it can easily be proven that the two Gauss-Seidel matrices $Q_G$ and $\bar{Q}_G$ are equal, that is, $Q_G = \bar{Q}_G$. Since $\bar{A}$ is a symmetric positive definite matrix, by theorem 4.4.32 of [1] the greedy algorithm converges.

### IV. Asynchronous Greedy Algorithms

The most important aspect of the implementation of the previous algorithm in a real network environment is the fact that each user needs to know only its cost function, the service rate and the throughputs of the other users. There is no need to know either the cost functions or the policy of the other classes. Thus there is no need for a central controller. But decentralization is not complete in the sense that every time a user computes its update it has to wait till the other users have finished their updates. Thus seeking an asynchronous algorithm is natural. The asynchronous version of this algorithm was first studied by Bovopoulos and Lazar in [2]. It was proved that the asynchronous algorithm does not converge for $n \geq 3$ under the assumption that $\beta_i$'s are equal for all $i$. In this section we present the asynchronous scheme following closely [2] and then give necessary and sufficient conditions for the convergence in more general cases.

Note that the equation $Ax = b$ can be reduced to the fixed point equation

$$x = Q_Jx + b$$

where

$$Q_J = I - D^{-1}\bar{A} = \begin{pmatrix}
0 & -a_1 & -a_1 & \cdots & -a_1 \\
-a_2 & 0 & -a_2 & \cdots & -a_2 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
-a_n & -a_n & \cdots & \cdots & 0
\end{pmatrix}.$$ 

We have put $a_i = \beta_i/(\beta_i + 1)$ for $i = 1, 2, \ldots, n$. The matrix $Q_J$ is the Jacobi matrix.

In [7], Chazan and Miranker showed that the Jacobi iterative technique can be performed asynchronously and called it a chaotic relaxation method. By asynchronous it is meant that all users may communicate with the given user the information with variable but bounded delays. At each time instant a component is updated using all the available information at that instant. Certain conditions on the maximum allowed delays in the calculations have to be fulfilled to ensure convergence. Let $\rho(Q_J)$ denote the spectral radius of matrix $Q_J$ and $|Q_J|$ be the matrix with entries $|q_{ij}|$, i.e., the absolute values of the entries of $Q_J$. The following theorem, quoted without proof from [7], provides the main result in the theory of chaotic relaxation.

**Theorem 2:** The sequence of iterates converges iff $\rho(|Q_J|) < 1$.

The convergence of the asynchronous scheme for $n = 2$ is given in [2]. Necessary and sufficient conditions for the convergence for the case $n = 3$ or the case when all the $\beta_i$'s are equal are derived in [9]. In the following we obtain a necessary and sufficient condition for general $\beta_i$'s and arbitrary $n$. To this end, we need the following two Lemmas.

**Lemma 1:** A necessary and sufficient condition for $\rho(|Q_J|) < 1$ is that both matrices $\bar{A}$ and $2D - \bar{A}$ are positive definite.

**Proof:** See the Appendix.

**Lemma 2:** A necessary and sufficient condition for $2D - \bar{A}$ to be positive definite is

$$\sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} > n - 2 \quad (1)$$

**Proof:**

$$\det(2D - \bar{A}) = \det \begin{pmatrix}
\frac{1 + \beta_1}{\beta_1} & -1 & \cdots & -1 \\
-1 & \frac{1 + \beta_2}{\beta_2} & \cdots & -1 \\
\vdots & \vdots & \ddots & \vdots \\
-1 & -1 & \cdots & \frac{1 + \beta_n}{\beta_n}
\end{pmatrix}$$

$$= \det \begin{pmatrix}
\frac{1 + \beta_1}{\beta_1} & -\frac{2}{\beta_1} & \cdots & -\frac{2}{\beta_1} \\
-\frac{1}{\beta_1} & 2 + \frac{1}{\beta_1} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
-1 & 0 & \cdots & 2 + \frac{1}{\beta_n}
\end{pmatrix}$$

$$= \det \begin{pmatrix}
\frac{1 + \beta_1}{\beta_1} & -\frac{2}{\beta_1} & \cdots & -\frac{2}{\beta_1} \\
0 & 2 + \frac{1}{\beta_1} & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 2 + \frac{1}{\beta_n}
\end{pmatrix}$$

$$= \left(1 + \frac{1}{\beta_1} - \frac{2}{\beta_1} + \frac{1}{\beta_1} \sum_{i=2}^{n} \frac{1}{\beta_i} \right) \prod_{i=2}^{n} \left(2 + \frac{1}{\beta_i} \right)$$
= \frac{1}{2\beta_1} \prod_{j=2}^{n} \left( 2 + \frac{1}{\beta_j} \right) 
abla \left( \sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} - (n - 2) \right). (2)

The second equality is obtained by subtracting the first column from the second to the nth column. The second equality is derived by multiplying the ith column \((i \geq 2)\) by \(1/(2\beta_i + 1)\) and adding the resulting column to the first column. If \(2\mathbf{D} - \mathbf{A}\) is positive definite, then

\[
\det(2\mathbf{D} - \mathbf{A}) > 0
\]

which implies that

\[
\sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} - (n - 3), \quad 1 \leq i_0 \leq n
\]

because the minor has the same structure as the matrix \(2\mathbf{D} - \mathbf{A}\). Since

\[
\sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} - (n - 3) = \sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} - (n - 2) + 1 - \frac{1}{1 + 2\beta_{i_0}} > 0
\]

we conclude that the minor determinant of order \(n - 1\) is positive. The positiveness of all other minor determinants can be easily deduced. Therefore, \(2\mathbf{D} - \mathbf{A}\) is positive definite.

Theorem 3: A necessary and sufficient condition for the asynchronous greedy algorithm to converge is

\[
\sum_{i=1}^{n} \frac{1}{1 + 2\beta_i} > n - 2, \quad n \geq 2.
\]

Proof: It follows directly from Theorem 2 and Lemmas 1-2.

As a check, for \(n = 3\), (5) reduces to

\[1 + \beta_1 + \beta_2 + \beta_3 - 4\beta_1\beta_2\beta_3 > 0\]

which is the condition presented in [9].

V. CONCLUSION

The Nash equilibrium has been presented as the appropriate operating point in a noncooperative multiclass network environment. The convergence of a decentralized synchronous greedy flow control algorithm was shown for the general case of \(n\) users in the single queue case. In the asynchronous case a necessary and sufficient condition for the convergence of the algorithm has been derived. Extensions to the general case of networks are under study. Relaxation techniques to speed up the convergence of the algorithms are also under investigation.

APPENDIX

Proof of Lemma 1: We follow the argument used in [6, p. 41] to prove Lemma 1. The Jacobi matrix \(Q_j\) is given by

\[
Q_j = I - A = I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}} = D^{-\frac{1}{2}} \left( I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}} \right) D^{-\frac{1}{2}}.
\]

Since \(A\) is symmetric, so is the matrix \(I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\), The latter is similar to \(Q_j\), hence, all the eigenvalues of \(Q_j\) are real. Since \(|Q_j| = -Q_j\), \(\text{tr}(Q_j) = 0\).

To prove the necessity, suppose \(\text{tr}(Q_j) < 1\). The eigenvalues of \(D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\) must lie in the interval \((0, 2)\), therefore, \(D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\) (and hence \(A\)) is positive definite. On the other hand, the eigenvalues of the matrix \(2I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\) are positive real numbers, therefore, \(2I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\) is positive definite, that is, \(2\mathbf{D} - \mathbf{A}\) is positive definite.

The sufficiency can be proven as follows. If \(\mathbf{A}\) and \(2\mathbf{D} - \mathbf{A}\) are positive definite, the eigenvalues of \(I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}}\) are less than 1, that is, the eigenvalues of \(Q_j\) are less than 1. On the other hand, that \(2\mathbf{D} - \mathbf{A}\) being positive definite implies that the eigenvalues of \(-\left( I - D^{-\frac{1}{2}} \mathbf{A} D^{-\frac{1}{2}} \right) = I - D^{-\frac{1}{2}} (2\mathbf{D} - \mathbf{A}) D^{-\frac{1}{2}}\) are less than 1, that is, the eigenvalues of \(-Q_j\) are less than 1. Hence \(\text{tr}(Q_j) < 1\).

REFERENCES


