Calcolo
DOI 10.1007/s10092-017-0214-7

On the solution of a rational matrix equation arising in G-networks

Beatrice Meini[1] · **Tommaso Nesti**[2]

Received: 1 June 2016 / Accepted: 4 January 2017

Abstract We consider the problem of solving a rational matrix equation arising in the solution of G-networks. We propose and analyze two numerical methods: a fixed point iteration and the Newton–Raphson method. The fixed point iteration is shown to be globally convergent with linear convergence rate, while the Newton method is shown to have a local convergence, with quadratic convergence rate. Numerical experiments show the effectiveness of the proposed methods.

Keywords Nonlinear matrix equation · Fixed point iteration · Newton–Raphson method · G-networks

Mathematic Subject Classification 65H10 · 65F30 · 60J22 · 15A24

1 Introduction

G-networks are a class of queueing networks introduced by Gelenbe in [7], originally inspired by the spiking behaviour of biophysical neurons. In some papers, G-networks

This work was partially supported by the project PRA 2015 "Mathematical models and computational methods for complex networks" of the University of Pisa, coordinated by Antonio Frangioni. B. Meini is also partially supported by GNCS of INdAM.

✉ Beatrice Meini
beatrice.meini@unipi.it

Tommaso Nesti
T.Nesti@cwi.nl

1 Dipartimento di Matematica, Università di Pisa, Pisa, Italy

2 Centrum Wiskunde & Informatica (CWI), Amsterdam, The Netherlands

Published online: 16 January 2017

are referred to as Random Neural Networks, and both the terminologies are currently used [7–9,18]. The novelty of G-networks, compared with standard queueing models, lies in the presence of negative customers, which have the capability to destroy usual customers and to disappear immediately after. G-networks have been applied in a variety of areas including image processing [2,17], combinatorial optimisation [10], and communication systems [5,14,16]. The usefulness of G-networks for these applications stems from their ability to learn from examples [9]. Learning algorithms require at each step the computation of the steady-state distribution π of the number of customers in the network, so it is important to develop efficient numerical methods for computing π.

The steady-state distribution π of a G-network can be expressed through the solution of a system of rational equations. More specifically, given an integer $N \geq 1$, which represents the number of queues, and given suitable parameters of the G-network $\mu_j, p_{ij}^+, p_{ij}^-, \Lambda_i^+, \Lambda_i^- \in \mathbb{R}^+$, for $i, j = 1, \ldots, N$, the problem is reduced to solving the system of rational equations

$$\begin{cases} \lambda_i^+ = \Lambda_i^+ + \sum_{j=1}^{N} \mu_j q_j p_{ji}^+, \\ \lambda_i^- = \Lambda_i^- + \sum_{j=1}^{N} \mu_j q_j p_{ji}^-, \quad i = 1, \ldots, N, \\ q_i = \dfrac{\lambda_i^+}{\mu_i + \lambda_i^-} \end{cases} \tag{1}$$

where λ_i^+, λ_i^-, for $i = 1, \ldots, N$ are the unknowns. When the system (1) admits a solution such that $\lambda_i^+, \lambda_i^- \in \mathbb{R}^+$ and $0 < q_i < 1$, for $i = 1, \ldots, N$, the network is stable and the steady-state distribution π can be expressed in product form, through the solution $\lambda_i^+, \lambda_i^-, i = 1, \ldots, N$.

Our goal is to compute the solution $\lambda_i^+, \lambda_i^-, i = 1, \ldots, N$ of (1) such that $\lambda_i^+, \lambda_i^- \in \mathbb{R}^+$ and $0 < q_i < 1$, for $i = 1, \ldots, N$.

We rewrite the system (1) in matrix form, yielding the equivalent rational matrix equation

$$z = T(z) := \Lambda^+ \left(D_z - P^+\right)^{-1} P^- D_\mu^{-1} + \alpha D_\mu^{-1} \tag{2}$$

where $z \in \mathbb{R}^{1\times N}$ is the row vector of unknowns, $\Lambda^+, \alpha, \mu \in \mathbb{R}^{1\times N}$ are suitable given row vectors, $P^+, P^- \in \mathbb{R}^{N\times N}$ are given nonnegative matrices and, for a given N dimensional vector w, $D_w = \mathrm{diag}(w) \in \mathbb{R}^{N\times N}$ is the diagonal matrix with the vector w on the main diagonal. The solution of interest is a vector $z \in \mathcal{D}$, where $\mathcal{D} = \{z \in \mathbb{R}^{1\times N} \ : \ z \geq \mathbf{1}\}$, $\mathbf{1} = [1, \ldots, 1] \in \mathbb{R}^{1\times N}$, and such that the stability condition

$$q := \Lambda^+ \left(D_z - P^+\right)^{-1} D_\mu^{-1} < \mathbf{1} \tag{3}$$

is satisfied, where the inequalities are applied component-wise. When the G-network is stable, there exists a unique solution $z^* \in \mathcal{D}$ to (2) which satisfies the stability condition (3) (see [9]).

We assume that the G-network is stable, and we propose and analyze two algorithms for computing the solution of (2) which satisfies the stability condition (3).

The first algorithm is a *fixed point iteration*, which generates the sequence of vectors $z^{(k+1)} = T(z^{(k)})$, for $k \geq 0$, starting from $z^{(0)} = \mathbf{1}$. We prove that this sequence converges to the sought fixed point z^*, with a linear rate of convergence given by the spectral radius of the Jacobian matrix of the function T at z^*. These properties are proved by using the theory of nonnegative matrices [3]. Properties of nonnegative matrices allow also to prove that the subsequences $(z^{(2k)})_{k\geq 0}$ and $(z^{(2k+1)})_{k\geq 0}$ of even and odd indices, respectively, satisfy

$$z^{(2k)} \leq z^{(2(k+1))} \leq z^* \leq z^{2(k+1)+1} \leq z^{(2k+1)}, \quad k = 0, 1, \ldots, \tag{4}$$

i.e., the convergence is alternate around the fixed point. Therefore the difference between two subsequent vectors yields an upper bound for the error at each step. Moreover, each iteration requires the inversion of an M-matrix and operations among nonnegative vectors and matrices, therefore all the computations are numerically stable.

The second algorithm is obtained by applying the *Newton-Raphson* method to the equation $S(z) := T(z) - z = 0$. Newton's method has proven to be useful in a variety of problems related to Markov chains, such as [4,11,13]. In this context, the method generates the sequence of vectors

$$z^{(k+1)} = z^{(k)} - (T(z^{(k)}) - z^{(k)})(T'(z^{(k)}) - I)^{-1}, \quad k = 0, 1, \ldots, . \tag{5}$$

starting from an initial approximation $z^{(0)}$, where $T'(z)$ is the Jacobian matrix of the function T at z. We prove that the iteration (5) is well defined and locally convergent to the fixed point z^*, with a quadratic rate of convergence.

We compare these two methods with the algorithm proposed in [6], which to the best of our knowledge is the standard method used in learning algorithms involving G-networks [18]. From the numerical experiments, the Newton-Raphson iteration is preferable for moderate values of N. This property makes the Newton-Raphson algorithm an advisable choice for applications where the steady-state distributions of many moderate-sized (usually $N < 100$) G-networks have to be computed, which is often the case in Communication System applications [5,14].

The paper is organized as follows. In Sect. 2 we specify the notation and recall some basic results and definitions. In Sect. 3 we describe the G-network model and the main existing results. In Sect. 4 we formulate the matrix-form of equations (1) and we study several properties that will be used in Sect. 5, where we develop and analyze the two new numerical methods for the computation of solution of (1). Section 6 is devoted to numerical experiments.

2 Nonnegative matrices

In this section we recall some properties of nonnegative matrices that will be used throughout the paper. For the results reported here we refer the reader to the books [3] and [19].

A matrix $A = (a_{ij})_{ij} \in \mathbb{R}^{N\times N}$ is said to be nonnegative (positive), and we will write $A \geq 0$ ($A > 0$), if $a_{ij} \geq 0$ ($a_{ij} > 0$) for all $i, j = 1, \ldots, N$.

Denote by $\rho(A)$ the spectral radius of a matrix A.

Theorem 1 *(Perron–Frobenius) Let $A \in \mathbb{R}^{N\times N}$, $A \geq 0$, be an irreducible matrix. Then the following properties hold:*

1. *$\rho(A) > 0$ and it is an eigenvalue of A;*
2. *There exists a vector $v > 0$ such that $Av = \rho(A)v$;*
3. *If $B \geq A$ and $B \neq A$, then $\rho(B) > \rho(A)$;*
4. *$\rho(A)$ is a simple eigenvalue.*

A useful corollary is the following:

Corollary 1 *Let $A \geq 0$ be an irreducible matrix. Then:*

1. *If the row sums of A are constant, i.e. $\sum_{j=1}^{N} a_{ij} = \sigma \; \forall i = 1, \ldots, N$, then $\rho(A) = \sigma$.*
2. *If the row sums of A have a minimum $\underline{\sigma}$ and a maximum $\overline{\sigma}$, then $\underline{\sigma} < \rho(A) < \overline{\sigma}$.*

Let $B \in \mathbb{R}^{N\times N}$, $B \geq 0$ and let $s \in \mathbb{R}$. The matrix $A = sI - B$ is said to be an M-matrix if $\rho(B) \leq s$, and it is a non-singular M-matrix if $\rho(B) < s$. A matrix $A \in \mathbb{R}^{N\times N}$ is said a Z-matrix if $a_{ij} \leq 0$ for all $i \neq j$.

It is clear that an M-matrix is also a Z-matrix. Regarding the opposite implication, the following theorem provides useful criteria for a Z-matrix to be an M-matrix.

Theorem 2 *Let $A \in \mathbb{R}^{N\times N}$ be a Z-matrix. Then A is a non-singular M-matrix if and only if one of the following properties holds:*

1. *The eigenvalues of A have positive real part;*
2. *A is non-singular and A^{-1} is non-negative;*
3. *There exists a vector $x > 0$ such that $Ax > 0$.*

We conclude this section with the following theorem.

Theorem 3 *Let $A = M - N$ be a regular splitting of the matrix A, i.e.* det $M \neq 0$, *$M^{-1} \geq 0$ and $N \geq 0$. Then, A is nonsingular with $A^{-1} \geq 0$ if and only if $\rho(M^{-1}N) < 1$, where*

$$\rho(M^{-1}N) = \frac{\rho(A^{-1}N)}{1 + \rho(A^{-1}N)}.$$

3 The model and the matrix equation

The basic G-network model consists of an open network of N queues in which two types of customers circulate: positive and negative ones. Each queue consists of one server with independent exponentially distributed service times, infinite waiting room and First In First Out (FIFO) policy for positive customers.

Positive customers obey standard service and routing disciplines as in conventional queueing network models. Upon their arrival on a queue, if the server is idle they immediately start being served, otherwise they queue, thus increasing the waiting line length.

Negative customers behave in the following way: when a negative customer joins a non-empty queue, it destroys one of the present positive customers (in the case of FIFO policy, the destroyed positive customer will be the one who arrived last at that queue). If the queue is empty, the negative customer simply vanishes without doing anything else. Negative customers are not stored in the queue and they will disappear as soon as they have accomplished their task: as a result, they can not be observed, only the effect of their arrivals can. Finally, negative customers actions are supposed to be taken instantaneously.

Upon completion of service in queue i, the newly served customer either reaches queue j as a positive customer with probability p_{ij}^{+}, or as a negative customer with probability p_{ij}^{-}, or it departs from the network with probability d_i. It is important to note that positive customers leaving a queue can become negative when they visit the next queue. These probabilities must sum up to one yielding

$$\sum_{j=1}^{N}\left(p_{ij}^{+}+p_{ij}^{-}\right)+d_i=1, \quad i=1,\ldots,N. \tag{6}$$

Let $p_{ij}=p_{ij}^{+}+p_{ij}^{-}$ for $i,j=1,\ldots,N$. The matrix $P=(p_{ij})\in\mathbb{R}^{N\times N}$ represents the movement of customers between queues.

Customers leaving a queue are not allowed to return directly back to the same queue, i.e. $p_{ii}=0$ for all i. Let $P^{+}=(p_{ij}^{+})\in\mathbb{R}^{N\times N}$ and $P^{-}=(p_{ij}^{-})\in\mathbb{R}^{N\times N}$. The matrices P^{+} and P^{-} are nonnegative, with zero diagonal entries and such that $P=P^{+}+P^{-}$ is row substochastic, i.e.

$$\sum_{j=1}^{N} p_{ij}\leq 1 \quad i=1,\ldots,N.$$

We assume also that at least for one row index i the inequality in the above formula is strict, and that P is irreducible.

Finally, positive and negative customers can also arrive to queue i from the outside world according to independent Poisson processes with rates Λ_i^{+} and Λ_i^{-} respectively. We assume that these processes are independent of each other. To avoid trivial cases, we suppose that both $\Lambda^{+}=[\Lambda_1^{+},\ldots,\Lambda_N^{+}]$ and $\Lambda^{-}=[\Lambda_1^{-},\ldots,\Lambda_N^{-}]$ are different from the zero vector. Finally, we denote by $\mu_i\in(0,+\infty)$ the service rate of the

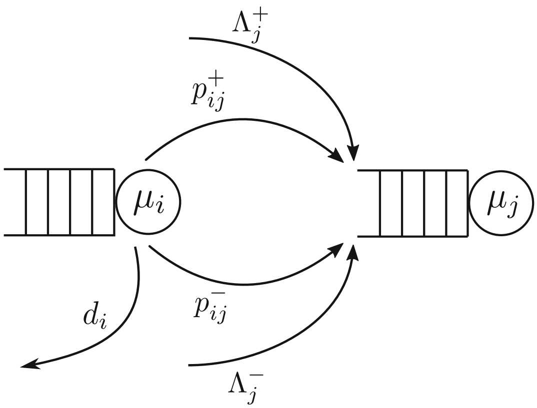

Fig. 1 The basic G-network model

queue i, meaning that the service time distribution of the single server in queue i has probability density function

$$\phi(x) = \begin{cases} \mu_i e^{-\mu_i x} & x \geq 0 \\ 0 & x < 0. \end{cases}$$

Figure 1 represents the traffic between two queues i and j.

From a neural network perspective, positive customers represent *excitation* and negative customers represent *inhibition* of a queue, which is usually called a *neuron* in this setting. The number of positive customers at a neuron, which is a non-negative integer, represents the *potential* of that neuron.

The state of queue i at time $t \in (0, +\infty)$ is described by the random variable $K_i(t)$, with support $\mathbb{N}$, representing the number of customers present in queue i at time t. These customers are necessarily positive customers, since negative customer, by definition, are never stored in a queue.

The state of the network at time $t \in (0, +\infty)$ is described by the random vector $K(t) = (K_1(t), \ldots, K_N(t))$, with support $\mathbb{N}^N$.

Letting $\pi(k_i, t) = \mathbb{P}(K_i(t) = k_i)$ and $\pi(k, t) = \mathbb{P}(K(t) = k)$, for $k_i \in \mathbb{N}$ and $k \in \mathbb{N}^N$, we are interested in determining, when they exist, the *steady-state* (or *stationary*) probability distributions for the queues state $\pi(k_i) = \lim_{t\to+\infty} \pi(k_i, t)$ and for the network state $\pi(k) = \lim_{t\to+\infty} \pi(k, t)$.

To this regard, an important role is played by the the system of non-linear equations

$$\begin{cases} x_i = \Lambda_i^+ + \sum_{j=1}^{N} \mu_j \frac{x_j}{\mu_j + y_j} p_{j,i}^+ \\ y_i = \Lambda_i^- + \sum_{j=1}^{N} \mu_j \frac{x_j}{\mu_j + y_j} p_{j,i}^- \quad i = 1, \ldots, N, \end{cases} \tag{7}$$

in the unknowns $x_i, y_i, i = 1, \ldots, N$.

The main result regarding the stationary distribution is given in the following theorem, proven by Gelenbe in [7].

Theorem 4 *If the system* (7) *admits a unique solution* λ_i^+, λ_i^-, $i = 1, \ldots, N$ *such that* $\lambda_i^+, \lambda_i^- > 0$ *and* $0 < q_i < 1$ *for any* $i = 1, \ldots, N$, *with* $q_i = \frac{\lambda_i^+}{\mu_i + \lambda_i^-}$, *then the stationary distributions* $\pi(k_i)$ *and* $\pi(k)$ *exist and are given by*

$$\pi(k_i) = (1 - q_i) q_i^{k_i}, \quad \pi(k) = \prod_{i=1}^{N} \pi(k_i). \tag{8}$$

The above theorem states that if the system of nonlinear equations (7) admits a positive solution with $0 < q_i < 1$ for all $i = 1, \ldots, N$, then the stationary distribution of the network state exists and it is given as a product form of the stationary distribution of each queue. In this case the network is said to be *stable* and q_i represents the stationary probability that the number of customers in queue i is positive. Moreover, the solution λ_i^+, λ_i^-, $i = 1, \ldots, N$, represents the mean arrival rates of positive and negative customers, respectively, to queue i, in steady-state.

Concerning the existence and uniqueness of the solution of (7), we have the following results [7,9], that can be proved by applying the Brower theorem.

Theorem 5 *A non-negative solution* λ_i^+, λ_i^-, $i = 1, \ldots, N$, *to Eq.* (7) *always exists. If a positive solution* λ_i^+, λ_i^- *to Eq.* (7) *exists with* $0 < q_i < 1$ *for any* $i = 1, \ldots, N$, *then it is the unique solution.*

By following [7], the system of equations (7) can be written in a more compact form. Define the row vectors

$$\begin{aligned} x &= [x_1, \ldots, x_N] \\ y &= [y_1, \ldots, y_N] \\ \Lambda^+ &= \left[\Lambda_1^+, \ldots, \Lambda_N^+\right] \\ \Lambda^- &= \left[\Lambda_1^-, \ldots, \Lambda_N^-\right] \end{aligned}$$

and let $D_{f(y)}$ be the $N \times N$ diagonal matrix with diagonal elements $f_j(y) = \frac{\mu_j}{\mu_j + y_j}$, $j = 1, \ldots, N$. Equation (7) can be written as

$$\begin{aligned} x\left(I - D_{f(y)} P^+\right) &= \Lambda^+ \\ y &= \Lambda^- + x D_{f(y)} P^-. \end{aligned} \tag{9}$$

As P is irreducible and $\rho(P) < 1$, from the Perron–Frobenius Theorem 1, it follows from $P^+ \leq P$, $P_- \leq P$ and $P^+ \neq P$, $P^- \neq P$, that $\rho(P^+) < 1$ and $\rho(P^-) < 1$. We restrict our attention to the case where $x \geq 0$ and $y \geq 0$, therefore in particular $0 \leq f_j(y) \leq 1$. Since $P \geq 0$ and $0 \leq f_j(y) \leq 1$, we have $0 \leq D_{f(y)} P^+ \leq P^+$. According to the Perron–Frobenius Theorem 1, we get $\rho(D_{f(y)} P^+) \leq \rho(P^+) < 1$.

Therefore matrix $I - D_{f(y)}P^+$ is non-singular and we may recover y from (9) and write

$$y = \Lambda^+ \left(I - D_{f(y)}P^+\right)^{-1} D_{f(y)}P^- + \Lambda^-. \tag{10}$$

Consider now the row vector $w = y - \Lambda^-$, $w = (w_j)_{j=1,\ldots,N}$, so that $f_j(y) = \frac{\mu_j}{\mu_j+\Lambda_j^-+w_j}$, $j = 1, \ldots, N$. Define

$$D_{r(w)} = \mathrm{diag}(r_1(w), \ldots, r_N(w)), \quad r_j(w) = \frac{\mu_j}{\mu_j + \Lambda_j^- + w_j}, \; j = 1, \ldots, N.$$

In view of (9) and (10), the system (7) can be written in the equivalent fixed-point form

$$w = G(w)$$

where $G : \mathbb{R}^{1\times N} \to \mathbb{R}^{1\times N}$ is given by

$$G(w) = \Lambda^+ \left(I - D_{r(w)}P^+\right)^{-1} D_{r(w)}P^- \tag{11}$$

and $y = w + \Lambda^-$. We are interested in the nonnegative solutions w, since, in the solution of interest, the vector $y = (y_j)_j$ represents the mean arrival rate $\lambda^- = (\lambda_j^-)_j$ of negative customers and we have $\lambda^- - \Lambda^- \geq 0$.

4 The rational matrix equation

In this section we manipulate the equation $w = G(w)$, where G is defined in (11). This way, we obtain a different formulation of the matrix equation, which will be useful to study convergence properties of iterative methods for its solution.

Since $r_j(w) > 0$ for $j = 1\ldots, N$, the diagonal matrix $D_{r(w)}$ is non-singular and we can write $G(w) = \Lambda^+(D_{r(w)}^{-1} - P^+)^{-1}P^-$. Set $D_z = D_{r(w)}^{-1}$, so that $D_z = \mathrm{diag}(z)$ and

$$z_j = \frac{1}{r_j(w)} = \frac{\mu_j + y_j}{\mu_j} = \frac{\mu_j + \Lambda_j^- + w_j}{\mu_j} = \frac{\alpha_j + w_j}{\mu_j}, \tag{12}$$

where $\alpha_j = \mu_j + \Lambda_j^- > 0$. In this way, we have $w = zD_\mu - \alpha$, where $\alpha = (\alpha_j)$ and $D_\mu = \mathrm{diag}(\mu)$.

Therefore, the equation $w = G(w)$ can be rewritten equivalently as

$$zD_\mu - \alpha = \Lambda^+ \left(D_z - P^+\right)^{-1} P^-,$$

i.e., in the form $z = T(z)$, where

$$T(z) = \Lambda^+ \left(D_z - P^+\right)^{-1} P^- D_\mu^{-1} + \alpha D_\mu^{-1}, \tag{13}$$

and where the function $T : \mathbb{R}^{1\times N} \to \mathbb{R}^{1\times N}$ is well defined for $z \in \mathcal{D}$, where $\mathcal{D} = \{z \in \mathbb{R}^{1\times N} \ : \ z \geq \mathbf{1}\}$. The variables w and z and the functions G and T satisfy the relations

$$w = zD_\mu - \alpha, \tag{14a}$$

$$G(w) = T(z)D_\mu - \alpha. \tag{14b}$$

The results of Theorem 5 immediately translate in this new formulation, yielding the existence of a fixed point $z^* = (w^* + \alpha)D_\mu^{-1} \geq \mathbf{1}$ for the function $T(z)$ in $z \in \mathcal{D}$, where $w^* = \lambda^- - \Lambda^-$.

Observe that for any non-negative matrix P with $\rho(P) < 1$ and for any $z \in \mathcal{D}$, the matrix $D_z - P$ is a nonsingular M-matrix, since it is a Z-matrix and $0 \leq D_z^{-1}P \leq P$, so that $\rho(D_z^{-1}P) \leq \rho(P) < 1$, whence $D_z - P$ is invertible and $(D_z - P)^{-1} = (I - D_z^{-1}P)^{-1}D_z^{-1} \geq 0$.

In particular, if $z \in \mathcal{D}$, the matrices $D_z - P^+$, $D_z - P^-$ and $D_z - P^+ - P^-$ are non-singular M-matrices. Moreover, if $w, z \in \mathbb{R}^{1\times N}$ are such that $w, z \in \mathcal{D}$, one has

$$\left(D_w - P^+\right)^{-1} - \left(D_z - P^+\right)^{-1} = \left(D_z - P^+\right)^{-1}(D_z - D_w)\left(D_w - P^+\right)^{-1}. \tag{15}$$

Therefore, the following expression for $T(w) - T(z)$ holds:

$$\begin{aligned} T(w) - T(z) &= \Lambda^+\left(D_z - P^+\right)^{-1}(D_z - D_w)\left(D_w - P^+\right)^{-1}P^-D_\mu^{-1} \\ &= \mathbf{1}\,\mathrm{diag}\left(\Lambda^+\left(D_z - P^+\right)^{-1}\right)\mathrm{diag}(z-w)\left(D_w - P^+\right)^{-1}P^-D_\mu^{-1} \\ &= (z-w)\,\mathrm{diag}\left(\Lambda^+\left(D_z - P^+\right)^{-1}\right)\left(D_w - P^+\right)^{-1}P^-D_\mu^{-1}. \end{aligned} \tag{16}$$

Proposition 1 *The function $T(z)$ of (13) satisfies the following properties:*

1. $\mathbf{1} \leq T(z) \leq T(\mathbf{1})$ *for any* $z \in \mathcal{D}$*;*
2. *if* $x, y \in \mathcal{D}$ *are such that* $x \leq y$ *then* $T(x) \geq T(y)$*, i.e.,* $T(z)$ *is monotonic non–increasing;*
3. $T(z)$ *is Lipschitz continuous in* $\mathcal{D}$*.*

Proof If $z \in \mathcal{D}$ then $D_z - P^+$ is a nonsingular M-matrix, therefore $(D_z - P^+)^{-1} \geq 0$. Since $\Lambda^+, P^-, \alpha, D_\mu^{-1} \geq 0$, we have

$$T(z) = \Lambda^+\left(D_z - P^+\right)^{-1}P^-D_\mu^{-1} + \alpha D_\mu^{-1} \geq \alpha D_\mu^{-1} = \mathbf{1} + \Lambda^-D_\mu^{-1} \geq \mathbf{1},$$

where the latter inequality follows from the fact that $\alpha = \mu + \Lambda^-$ and $\Lambda^- \geq 0$, $D_\mu^{-1} \geq 0$. The function $T(z)$ is monotonic non–increasing since the matrix $(D_z - P^+)^{-1} = (I - D_z^{-1}P^+)^{-1}D_z^{-1} = \sum_{n\geq 0}(D_z^{-1}P^+)^n D_z^{-1}$ is non–increasing. In particular, we have

$$\left(D_z - P^+\right)^{-1} \leq (I - P^+)^{-1} \quad \forall z \in \mathcal{D}, \tag{17}$$

yielding $T(z) \leq T(\mathbf{1})$ for $z \in \mathcal{D}$. Concerning the Lipscthiz continuity, from (16) and (17), we have $\|T(x) - T(y)\|_\infty \leq K\|x - y\|_\infty$ for a suitable costant K, which completes the proof. □

Equation (15) in particular implies that for any $z \in \mathcal{D}$

$$\lim_{\substack{w \to z \\ w \in \mathcal{D}}} \left\| \left(D_w - P^+\right)^{-1} - \left(D_z - P^+\right)^{-1} \right\| = 0 \tag{18}$$

since $(D_z - P^+)^{-1}$ is bounded for all $z \in \mathcal{D}$.

Theorem 6 *The function $T(z)$ of (13) is Fréchet differentiable for $z \in \mathcal{D}$ and its Fréchet derivative is*

$$T'(z) := -\mathrm{diag}\left(\Lambda^+ \left(D_z - P^+\right)^{-1}\right) \left(D_z - P^+\right)^{-1} P^- D_\mu^{-1}. \tag{19}$$

Moreover, $T'(z) \leq 0$, $T'(z) \leq T'(w)$ whenever $w, z \in \mathcal{D}$ and $z \leq w$, and the function $T'(z)$ is Lipschitz continuous for $z \in \mathcal{D}$.

Proof From (16), for any row vector $h \in \mathbb{R}^{1\times N}$ such that $z + h \in \mathcal{D}$, we have

$$T(z+h) - T(z) = -h\,\mathrm{diag}(\Lambda^+(D_z - P^+)^{-1})(D_{z+h} - P^+)^{-1} P^- D_\mu^{-1}.$$

Therefore, setting $A_z = (D_z - P^+)^{-1}$, we have

$$T(z+h) - T(z) - hT'(z) = h\,\mathrm{diag}\left(\Lambda^+ \left(D_z - P^+\right)^{-1}\right)(A_{z+h} - A_z) P^- D_\mu^{-1},$$

hence, from (17) and from the boundedness of $\Lambda^+(D_w - P^+)^{-1}$, we have

$$\lim_{h \to 0} \frac{\|T(z+h) - T(z) - hT'(z)\|_\infty}{\|h\|_\infty} = 0,$$

which shows that $T(z)$ is Fréchet differentiable and that $T'(z)$ given in (19) is its derivative. The matrix $T'(z)$ is nonpositive since all the involved matrices in (19) are nonnegative. Moreover, if $w, z \in \mathcal{D}$ and $z \leq w$, then $(D_z - P^+)^{-1} \geq (D_w - P^+)^{-1}$, hence $T'(z) \leq T'(w)$. Now we prove that $T'(z)$ is Lipschitz continuous. Let h be a nonnegative row vector. By means of simple computations we find that

$$\begin{aligned} &T'(z) - T'(z+h) \\ &\quad = -\Big(\mathrm{diag}(\Lambda^+ A_{z+h})(A_z D_h A_{z+h}) - \mathrm{diag}(\Lambda^+ A_z D_h A_{z+h}) A_z\Big) P^- D_\mu^{-1}. \end{aligned}$$

Taking norms, since A_z is bounded for $z \in \mathcal{D}$, yields

$$\|T'(z) - T'(z+h)\|_\infty \leq C\|h\|_\infty$$

for a suitable constant C, which completes the proof. □

The following result will allow us to study the convergence properties of the computational methods described in the next section.

Theorem 7 *Let $z \in \mathcal{D}$ be such that*

$$\Lambda^{+}\left(D_z - P^{+}\right)^{-1} D_\mu^{-1} < \mathbf{1}. \tag{20}$$

Then $\rho(T'(z)) < 1$ and $\rho(T'(w)) \le \rho(T'(z)) < 1$ for any $w \ge z$.

Proof By (20), $\mathrm{diag}(\Lambda^{+}(D_z - P^{+})^{-1}) \le D_\mu$. Consequently, from (19), we have

$$0 \le -T'(z) \le D_\mu \left(D_z - P^{+}\right)^{-1} P^{-} D_\mu^{-1},$$

therefore we have

$$\rho(T'(z)) \le \rho\left(D_\mu \left(D_z - P^{+}\right)^{-1} P^{-} D_\mu^{-1}\right) = \rho\left(\left(D_z - P^{+}\right)^{-1} P^{-}\right).$$

Since $D_z - P^{+} - P^{-}$ is a nonsingular M-matrix, then $\rho((D_z - P^{+})^{-1} P^{-}) < 1$ for Theorem 3. Since the matrix $-T'(z)$ is nonnegative and monotonic non–increasing in $\mathcal{D}$, for the Perron–Frobenius Theorem, we have $\rho(T'(w)) \le \rho(T'(z))$ if $w \ge z$. □

Recall that the condition for a G-network to be stable is $q_i = \frac{\lambda_i^{+}}{\mu_i + \lambda_i^{-}} < 1$ for all $i = 1, \ldots, N$. Since $z_i = \frac{\mu_i + \lambda_i^{-}}{\mu_i}$ from (12), we have $q_i = \frac{\lambda_i^{+}}{\mu_i + \lambda_i^{-}} = \frac{\lambda_i^{+}}{z_i \mu_i}$. In view of (9), we have

$$\lambda^{+} = \Lambda^{+}\left(I - D_{f(y)} P^{+}\right)^{-1} = \Lambda^{+}\left(I - D_z^{-1} P^{+}\right)^{-1}$$

therefore the row vector $q = (q_i)_i$ can be rewritten as

$$q = \Lambda^{+}\left(I - D_z^{-1} P^{+}\right)^{-1} D_z^{-1} D_\mu^{-1} = \Lambda^{+}\left(D_z - P^{+}\right)^{-1} D_\mu^{-1},$$

and the stability condition becomes

$$\Lambda^{+}\left(D_{z^*} - P^{+}\right)^{-1} D_\mu^{-1} < \mathbf{1}, \tag{21}$$

where z^* is the fixed point of the function $T(z)$ in $\mathcal{D}$. Therefore, we have the following corollary.

Proposition 2 *Suppose that the G-network is stable, and let z^* be the fixed point of the function $T(z)$ in $\mathcal{D}$. Then $\rho(T'(z^*)) < 1$.*

The above result, together with Theorem 7, implies that, if the G-network is stable, then $\rho(T'(z)) < 1$ for and $z \ge z^*$.

5 Numerical methods

In this section we present two numerical methods for solving the equation $z = T(z)$ in $\mathcal{D}$, with $T(z)$ given in (13). In the following we will assume that the G-network is stable. The stability condition implies that there exists a unique solution z^* in $\mathcal{D}$ of the equation $z = T(z)$, and the inequality $\Lambda^+(D_{z^*} - P^+)^{-1}D_\mu^{-1} < \mathbf{1}$ holds.

5.1 Fixed-point iteration

The first method we consider is the fixed point iteration:

$$\begin{cases} z^{(k+1)} = T(z^{(k)}), & k \geq 0, \\ z^{(0)} \in \mathbb{R}^{1\times N}. \end{cases} \tag{22}$$

In view of Proposition 1, if $z^{(0)} \in \mathcal{D}$, the iteration (22) is well defined and $z^{(k)} \in \mathcal{D}$ for any $k \geq 1$. Moreover $T(z) \neq \mathbf{1}$, since P^-, D_μ^{-1}, $(D_z - P^+)^{-1} \neq 0$ and Λ^+, Λ^- are different from zero. As a consequence, also the fixed point z^* is such that $z^* \geq \mathbf{1}$, $z^* \neq \mathbf{1}$.

We are now able to prove that the iteration (22) is locally convergent:

Theorem 8 *Suppose that the G-network is stable. Then:*

1. *The function $T : \mathbb{R}^{1\times N} \to \mathbb{R}^{1\times N}$ is contractive in a neighbourhood of z^*, i.e. there exists a norm $\|\cdot\|$ on $\mathbb{R}^{1\times N}$, a neighbourhood $I(z^*) \subset \mathcal{D}$ of z^* and a scalar $0 \leq \gamma < 1$ such that for all $z, w \in I(z^*)$*

$$\|T(z) - T(w)\| \leq \gamma\|z - w\|; \tag{23}$$

2. *For all $z^{(0)} \in I(z^*)$, the sequence (22) converges to z^*, which is the unique fixed point of T in $I(z^*)$.*

Proof Since $\rho(T'(z^*)) < 1$ (see Theorem 2), there exists a norm $\|\cdot\|$ on $\mathbb{R}^N$ such that the induced matrix norm satisfies $\|T'(z^*)\| < 1$. Since $z^* \geq \mathbf{1}$, by continuity of T' (see Theorem 6) there exists a compact neighbourhood $I(z^*) \subset \mathcal{D}$ of z^* , such that $\|T'(z)\| < 1 \quad \forall z \in I(z^*)$, so that $\gamma := \sup_{z\in I(z^*)} \|T'(z)\| < 1$. By using the mean value theorem we obtain (23). The local convergence and the uniqueness are straightforward applications of the Contraction Mapping Theorem [12, Chap. 7]. □

The above result states that the fixed point iteration (22) converges to the sought solution z^*, provided that the stability condition is satisfied. Indeed, the stability condition guarantees that the spectral radius of the Jacobian of $T(z^*)$ is less than one. If the stability condition is not satisfied, the sequence $\{z^{(k)}\}_k$ may or may not converge; if it converges, the limit is a solution of the equation $T(z) = z$ which does not verify the stability condition (21), and does not have a physical meaning. Therefore, in the case where we do not know a priori whether the system is stable or not, we may compute the sequence $\{z^{(k)}\}_k$ and, if it converges to a solution satisfying (21), then the

G-network is stable. However, in practice, some sufficient conditions for the network stability, which can be checked a priori, are available in the literature (see for example the concept of *damped network* in [8]).

Theorem 8 is a local convergence result: it states that, if $z^{(0)}$ is sufficiently close to z^*, then the sequence $\{z^{(k)}\}_k$ convergences to z^*. Instead, the following theorem provides a global convergence result, i.e., the choice $z^{(0)} = \mathbf{1}$ guarantees the convergence of the sequence $\{z^{(k)}\}_k$.

Theorem 9 *Assume that the G-network is stable. Take* $z^{(0)} = \mathbf{1}$ *and define*

$$H_k = -\mathrm{diag}\left(\Lambda^+\left(D_{z^*} - P^+\right)^{-1}\right)\left(D_{z^{(k)}} - P^+\right)^{-1} P^- D_\mu^{-1}.$$

Then the sequence (22) *satisfies the following properties:*

$$z^{(k)} - z^* = (z^{(0)} - z^*)H_0 H_1 \cdots H_{k-1}, \tag{24}$$

and

$$\mathbf{1} \le z^{(2k)} \le z^{(2(k+1))} \le z^* \le z^{(2(k+1)+1)} \le z^{(2k+1)}, \tag{25}$$

for $k \ge 0$. *Moreover* $\lim_{k\to\infty} z^{(k)} = z^*$ *and* $\lim_{k\to\infty} H_k = T'(z^*)$.

Proof From (16), we have

$$z^{(k)} - z^* = T(z^{(k-1)}) - T(z^*) = (z^{(k-1)} - z^*)H_{k-1},$$

therefore Eq. (24) follows by induction on k. Now we prove (25). First observe that, if $\mathbf{1} \le z \le w$ then $\mathbf{1} \le T(T(z)) \le T(T(w))$, since $T(z)$ in monotonic non–increasing and bounded thanks to Proposition 1. We show that inequality (25) holds for $k = 0$. Since $\mathbf{1} = z^{(0)} \le z^*$ then $\mathbf{1} \le T(T(z^{(0)})) \le T(T(z^*))$, i.e., $\mathbf{1} = z^{(0)} \le z^{(2)} \le z^*$. From the monotonicity of $T(z)$ it follows that $T(z^{(0)}) \ge T(z^{(2)}) \ge T(z^*)$, i.e., $z^* \le z^{(3)} \le z^{(1)}$. Therefore (25) holds for $k = 0$. We assume that it holds for $k \ge 0$ and we show that it is true for $k+1$. Since $\mathbf{1} \le z^{(2k)} \le z^{(2(k+1))} \le z^*$, from the monotonicity of $T(T(z))$ we have $\mathbf{1} \le T(T(z^{(2k)})) \le T(T(z^{(2(k+1))})) \le T(T(z^*))$, i.e., $\mathbf{1} \le z^{(2(k+1))} \le z^{(2(k+2))} \le z^*$. Similarly, we show that $z^* \le z^{(2(k+2)+1)} \le z^{(2(k+1)+1)}$. Since the sequences $\{z^{(2k)}\}_k$ and $\{z^{(2k+1)}\}_k$ are monotonic and bounded, there exist $\lim_{k\to\infty} z^{(2k)} = \underline{z}$ and $\lim_{k\to\infty} z^{(2k+1)} = \bar{z}$, with $\underline{z} = T(T(\underline{z}))$, $\bar{z} = T(T(\bar{z}))$ and $\underline{z} \le z^* \le \bar{z}$. We show that $\underline{z} = z^* = \bar{z}$. Let $w = \bar{z} - z^*$. In view of (16), one has

$$w = T(T(\bar{z})) - T(T(z^*)) = (T(z^*) - T(\bar{z}))R_1 = wR_2R_1,$$

with

$$R_1 = \mathrm{diag}\left(\Lambda^+\left(D_{z^*} - P^+\right)^{-1}\right)\left(D_{T(\bar{z})} - P^+\right)^{-1} P^- D_\mu^{-1},$$
$$R_2 = \mathrm{diag}\left(\Lambda^+\left(D_{\bar{z}} - P^+\right)^{-1}\right)\left(D_{z^*} - P^+\right)^{-1} P^- D_\mu^{-1}.$$

If $w \neq 0$, then w is an eigenvector of the matrix $R_2 R_1$, corresponding to the eigenvalue 1. Since $\mathbf{1} \leq z^* \leq \bar{z}$ and $\mathbf{1} \leq T(\bar{z}) \leq z^*$, we deduce that $R_1 \leq R$ and $R_2 \leq R$, where

$$R = \operatorname{diag}\left(\Lambda^+ \left(D_{z^*} - P^+\right)^{-1}\right)(I - P^+)^{-1} P^- D_\mu^{-1}.$$

Since $0 \leq R \leq (I - P^+)^{-1} P^-$, by using the same arguments of the proof of Theorem 7, we deduce that $\rho(R) < 1$. By the Perron–Frobenius Theorem we have $\rho(R_2 R_1) \leq \rho(R^2) < 1$, therefore w cannot be an eigenvector corresponding to the eigenvalue 1, hence $w = 0$. Finally, from continuity of the function $h(z) := -\operatorname{diag}(\Lambda^+(D_{z^*} - P^+)^{-1})(D_z - P^+)^{-1} P^- D_\mu^{-1}$ in $\mathcal{D}$, it follows that $\lim_{k\to\infty} H_k = T'(z^*)$. □

The alternating property stated by Theorem 9 is useful also from a practical standpoint, since it allows to upper bound the error of each component at each step. In fact, the following bound holds:

$$\left|z_i^{(k)} - z_i^*\right| \leq \left|z_i^{(k)} - z_i^{(k-1)}\right| \quad \forall i = 1, \ldots, N.$$

Therefore, the absolute value of the difference between the i-th component of two subsequent vectors is an upper bound to the approximation error of z_i^*.

The next theorem shows that the fixed point iteration (22) has a linear rate of convergence and the asymptotic reduction of the error is bounded by the spectral radius of the Jacobian $T'(z^*)$.

Theorem 10 *If $z^{(0)} = \mathbf{1}$ the sequence (22) satisfies*

$$\limsup_{k\to\infty} \sqrt[k]{\|z^{(k)} - z^*\|_\infty} \leq \rho(T'(z^*)). \tag{26}$$

Proof For $z \in \mathcal{D}$, let $h(z) = -\operatorname{diag}(\Lambda^+(D_{z^*} - P^+)^{-1})(D_z - P^+)^{-1} P^- D_\mu^{-1}$, so that $H_k = h(z^{(k)})$ and $T'(z^*) = h(z^*)$. Denote $J_* = T'(z^*)$. The function h is component-wise non-decreasing. Applying h to (25) yields

$$H_{2k-1} \leq H_{2k+1} \leq J_* \leq H_{2(k+1)} \leq H_{2k}, \quad k \geq 1. \tag{27}$$

Therefore, from (24), we obtain

$$z^{(k+1)} - z^* = (z^{(0)} - z^*) H_0 \ldots H_k = (z^{(0)} - z^*) H_0 \ldots H_{2(l-1)} H_{2l-1} H_{2l} \ldots H_k$$

where $0 \leq l < \lfloor k/2 \rfloor$ is a fixed nonnegative integer. Setting $w_l := (z^{(0)} - z^*) H_0 \ldots H_{2(l-1)}$, and thanks to (27), we obtain

$$z^{(k+1)} - z^* \leq w_l \underbrace{(J_* H_{2l})(J_* H_{2l}) \ldots (J_* H_{2l})}_{\lfloor k/2 \rfloor - l + 1 \text{ times}} = w_l (J_* H_{2l})^{\lfloor k/2 \rfloor - l + 1} S_k$$

where

$$S_k = \begin{cases} I & \text{if } k \text{ is even} \\ J_* & \text{if } k \text{ is odd.} \end{cases}$$

Taking norms yields

$$\|z^{(k+1)} - z^*\|_\infty \leq C_l \|(J_* H_{2l})^{\lfloor k/2 \rfloor + 1}\|_\infty$$

and

$$\sqrt[k]{\|z^{(k+1)} - z^*\|_\infty} \leq \sqrt[k]{C_l} \sqrt[k]{\|(J_* H_{2l})^{\lfloor k/2 \rfloor + 1}\|_\infty}$$

where

$$C_l = \|w_l\|_\infty \|S\|_\infty \|(J_* H_{2l})^{-l}\|_\infty.$$

Taking the limit for $k \to \infty$ yields

$$\limsup_{k\to\infty} \sqrt[k]{\|z^{(k+1)} - z^*\|_\infty} \leq \underbrace{\lim_{k\to\infty} \sqrt[k]{C_l}}_{=1} \lim_{k\to\infty} \sqrt[k]{\|(J_* H_{2l})^{\lfloor k/2 \rfloor + 1}\|_\infty} = \rho(J_* H_{2l})^{1/2}$$

where we used Gelfand's formula. Taking now the limit for $l \to \infty$ yields

$$\limsup_{k\to\infty} \sqrt[k]{\|z^{(k+1)} - z^*\|_\infty} \leq \lim_{l\to\infty} \rho(J_* H_{2l})^{1/2} = \rho(J_*^2)^{1/2} = \rho(J_*)$$

where we used the continuity of the spectral radius and of the function h. □

Concerning the computational cost, each iteration requires the computation of the vector $T(z^{(k)}) = \Lambda^+(D_{z^{(k)}} - P^+)^{-1} P^- D_\mu^{-1} + \alpha D_\mu^{-1}$. The matrices P^+, P^-, D_μ^{-1} and the vectors Λ^+, α are given parameters. The only costly operation that we have to perform at each step is the solution of the linear system $v(D_{z^{(k)}} - P^+) = \Lambda^+$, since the other multiplications are just vector-matrix products, yielding a cost of $O(N^3)$ time per step. Moreover, the matrix $D_{z^{(k)}} - P^+$ is a nonsingular M-matrix, and all the vectors involved in the iteration are nonnegative, therefore each step can be performed in a numerically stable way by exploiting the properties of M-matrices [1].

5.2 Newton–Raphson iteration

In this section we will focus on the equation $S(z) := T(z) - z = 0$. As usual we will suppose that the system is stable, therefore the unique fixed point z^* of T in $\mathcal{D}$ is the unique row vector in $\mathcal{D}$ such that $S(z) = 0$.

Since T is Fréchet differentiable in $\mathcal{D}$, also S is Fréchet differentiable in $\mathcal{D}$ and its derivative is $S'(z) = T'(z) - I$.

The Newton-Raphson iteration applied to the equation $S(z) = 0$ generates the sequence of row vectors

$$\begin{aligned} z^{(k+1)} &= z^{(k)} - S(z^{(k)})S'(z^{(k)})^{-1} \\ &= z^{(k)} - (T(z^{(k)}) - z^{(k)})(T'(z^{(k)}) - I)^{-1}, \quad k \geq 0. \end{aligned} \tag{30}$$

We can show that the function $S(z)$ is order–convex on the set $\mathcal{D} \subset \mathbb{R}^{1\times N}$, i.e.,

$$S(\lambda z + (1-\lambda)w) \leq \lambda S(z) + (1-\lambda)S(w),$$

whenever $z, w \in \mathcal{D}$ are comparable ($z \leq w$ or $w \leq z$) and $0 < \lambda < 1$. Unfortunately, $S'(z)^{-1}$ does not have in general sign properties, therefore we cannot immediately use the results of Sect. 13.3 of [15], to derive global convergence properties for the sequence (30).

However, under the stability condition, we can show that the sequence (30) is locally convergent, well defined and has a quadratic rate of convergence, as stated by the following result.

Theorem 11 *Suppose that the G-network is stable and let* $\|\cdot\|$ *be an operator norm. Then there exist* $\delta, M > 0$ *such that, for all* $z^{(0)} \in \mathcal{D}$ *such that* $\|z^{(0)} - z^*\| \leq \delta$, *the iteration* (30) *is well-defined,* $\lim_{k\to\infty} z^{(k)} = z^*$ *and*

$$\|z^{(k+1)} - z^*\| \leq M\|z^{(k)} - z^*\|^2 \quad \forall k \geq 0.$$

Proof Since $\rho(T'(z^*)) < 1$ by Theorem 7, $S'(z^*) = T'(z^*) - I$ is non-singular and, by Lemma 6, $S'(z)$ is also Lipschitz continuous for $z \in \mathcal{D}$. Therefore, there exists a neighbourhood $\mathcal{B} \subset \mathcal{D}$ of z^* such that $S'(z)$ is Lipschitz continuous, non-singular and such that $\|S'(z)^{-1}\|$ is uniformly bounded for any z belonging to $\mathcal{B}$. Therefore the thesis follows from general convergence properties of Newton's method (see Theorem 12.6.2 of [15]). □

Regarding the choice of the initial point $z^{(0)}$, we point out that in all the numerical experiments performed in Sect. 6 it was sufficient to choose $z^{(0)} = T(T(\mathbf{1}))$ to achieve convergence.

Concerning the computational cost, each step of the Newton-Raphson method requires the computation of the vector $T(z^{(k)}) = \Lambda^+(D_{z^{(k)}} - P^+)^{-1}P^-D_\mu^{-1} + \alpha D_\mu^{-1}$ and of the matrix $(I - T'(z^{(k)}))^{-1} = (I + \mathrm{diag}(\Lambda^+(D_{z^{(k)}} - P^+)^{-1})(D_{z^{(k)}} - P^+)^{-1}P^-D_\mu^{-1})^{-1}$. In practice it is convenient to first compute the inverse $(D_{z^{(k)}} - P^+)^{-1}$. Then we obtain $T(z^{(k)}) = \Lambda^+(D_{z^{(k)}} - P^+)^{-1}P^-D_\mu^{-1} + \alpha D_\mu^{-1}$ and $I - T'(z^{(k)}) = I + \mathrm{diag}(\Lambda^+(D_{z^{(k)}} - P^+)^{-1})(D_{z^{(k)}} - P^+)^{-1}P^-D_\mu^{-1}$ with just vector-matrix products, matrix-matrix product with a diagonal factor and matrix sums. The overall cost is $O(N^3)$ arithmetic operation, with an overall multiplicative constant greater than in the fixed point iteration. However, in general, the quadratic rate of convergence of Newton-Raphson makes it preferable to the fixed point iteration.

6 Numerical results

We compare the fixed point iteration and the Newton-Raphson method with the iterative method proposed by Forneau in [6], which, as far as we know, is the standard method used in learning algorithms involving G-networks.

We briefly recall this method. Consider, for each index $i = 1, \ldots, N$, the six sequences of real numbers

$$\{(\overline{q_i})_k\}_{k\geq 0}, \quad \{(\underline{q_i})_k\}_{k\geq 0}, \quad \left\{\left(\overline{\lambda_i^+}\right)_k\right\}_{k\geq 0}, \quad \left\{\left(\underline{\lambda_i^+}\right)_k\right\}_{k\geq 0}, \\ \left\{\left(\overline{\lambda_i^-}\right)_k\right\}_{k\geq 0}, \quad \left\{\left(\underline{\lambda_i^-}\right)_k\right\}_{k\geq 0}$$

defined by induction on $k \geq 0$ as follows:

$$\left(\overline{\lambda_i^+}\right)_k = \Lambda_i^+ + \sum_{j=1}^{N} \mu_j p_{j,i}^+ (\overline{q_j})_k \tag{31a}$$

$$\left(\underline{\lambda_i^+}\right)_k = \Lambda_i^+ + \sum_{j=1}^{N} \mu_j p_{j,i}^+ (\underline{q_j})_k \tag{31b}$$

$$\left(\overline{\lambda_i^-}\right)_k = \Lambda_i^- + \sum_{j=1}^{N} \mu_j p_{j,i}^- (\overline{q_j})_k \tag{31c}$$

$$\left(\underline{\lambda_i^-}\right)_k = \Lambda_i^- + \sum_{j=1}^{N} \mu_j p_{j,i}^- (\underline{q_j})_k \tag{31d}$$

$$(\overline{q_i})_{k+1} = \min\left(1, \left(\overline{\lambda_i^+}\right)_k / \left(\mu_i + \left(\underline{\lambda_i^-}\right)_k\right)\right) \tag{31e}$$

$$(\underline{q_i})_{k+1} = \min\left(1, \left(\underline{\lambda_i^+}\right)_k / \left(\mu_i + \left(\overline{\lambda_i^-}\right)_k\right)\right) \tag{31f}$$

with the following initial values:

$$(\overline{q_i})_0 = 1, \quad (\underline{q_i})_0 = 0. \tag{32}$$

For each $k \geq 0$, the iteration proceeds as follows: first compute the $(\overline{\lambda_i^+})_k$, $(\underline{\lambda_i^+})_k$, $(\overline{\lambda_i^-})_k$, $(\underline{\lambda_i^-})_k$ from the Eqs. (31a)–(31d), which only depends on the kwnon values $(\overline{q_i})_k$ and $(\overline{q_i})_k$, then compute $(\overline{q_i})_{k+1}$ and $(\overline{q_i})_{k+1}$ through (31e)–(31f). The following convergence result holds (see [6]):

Theorem 12 *Assume that, for any $i = 1, \ldots, N$, one of the two following assumption is satisfied:*

- *There is a strictly positive probability that a positive customer leaves the queue to go outside, i.e. $d_i > 0$;*

– *There is a strictly positive probability that a customer, either positive or negative, joins a queue j where the rate of negative customers coming from the outside is strictly positive, i.e.* $p_{ij}^{+} + p_{ij}^{-} > 0$ *and* $\Lambda_j^{-} > 0$.

Then the sequences $\{(\overline{q_i})_k\}_k$ *and* $\{(\underline{q_i})_k\}_k$ *defined in* (31a) *converge, from above and from below respectively, to* q_i, *where* $q_i = \frac{\lambda_i^+}{\mu_i+\lambda_i^-}$.

Regarding the convergence rate, it is proven in [6] that the sequences $(\overline{q_i})_k$ and $(\underline{q_i})_k$ are respectively upper and lower bounds for the q_i and that

$$\sum_{i=1}^{N} \mu_i[(\overline{q_i})_{k+1} - (\underline{q_i})_{k+1}] \le \epsilon \sum_{i=1}^{N} \mu_i[(\overline{q_i})_k - (\underline{q_i})_k] \tag{33}$$

where

$$\epsilon_i = \sum_{j=1}^{N} \left(p_{ij}^{+} + p_{ij}^{-}\right) \frac{\mu_j}{\mu_j + \Lambda_j^{-}}, \quad \epsilon = \max_{i=1,\dots,N} \epsilon_i < 1,$$

yielding that the sequence $\sum_{i=1}^{N} \mu_i[(\overline{q_i})_k - (\underline{q_i})_k]$ converges linearly to zero.

Let $(\overline{q}^{(k)})_{k\ge0}$, $(z_{FP}^{(k)})_{k\ge0}$, $(z_{NR}^{(k)})_{k\ge0}$ be the Fourneau (31a)–(31f), Fixed Point (22) and Newton-Raphson (30) iterations, respectively. The starting approximation for the Fourneau iteration is given by (32), for the Fixed Point we set $z^{(0)} = \mathbf{1}$ (see 9) and for the Newton-Raphson we set $z^{(0)} = T(T(\mathbf{1}))$.

In the following the three algorithms will be denoted by *FRN*, *FP*, *NR*. The relative errors for the three methods are denoted by

$$e_{FRN}^{(k)} = \frac{\|\overline{q}^{(k)} - \overline{q}^*\|_\infty}{\|\overline{q}^*\|_\infty}, \quad e_{FP}^{(k)} = \frac{\|z_{FP}^{(k)} - z_{FP}^*\|_\infty}{\|z_{FP}^*\|_\infty}, \quad e_{NR}^{(k)} = \frac{\|z_{NR}^{(k)} - z_{NR}^*\|_\infty}{\|z_{NR}^*\|_\infty} \tag{34}$$

for $k \ge 0$, where $\overline{q}^*$, z_{FP}^* and z_{NR}^* are the approximations to solutions computed by each of the three methods.

The algorithms have been implemented in Matlab ver. 7.5.0. We performed the experiments with a AMD Athlon II X 4 640, clockspeed 3, 0 GHz.

We generate a stochastic matrix A with uniformly distributed psuedo-random elements in the interval $(0, 1)$ and, for a given $x \in (0, 1)$, we set

$$\begin{aligned} D &= \mathrm{diag}(a_{11}, \dots, a_{NN}) \\ P^+ &= x(A - D) \\ P^- &= (1 - x)(A - D) \end{aligned}$$

so that $P^+ + P^- + D = A$ is stochastic. In this way P^+ and P^- are both full matrices, and varying x we control the internal flow of customers in the network: the larger x,

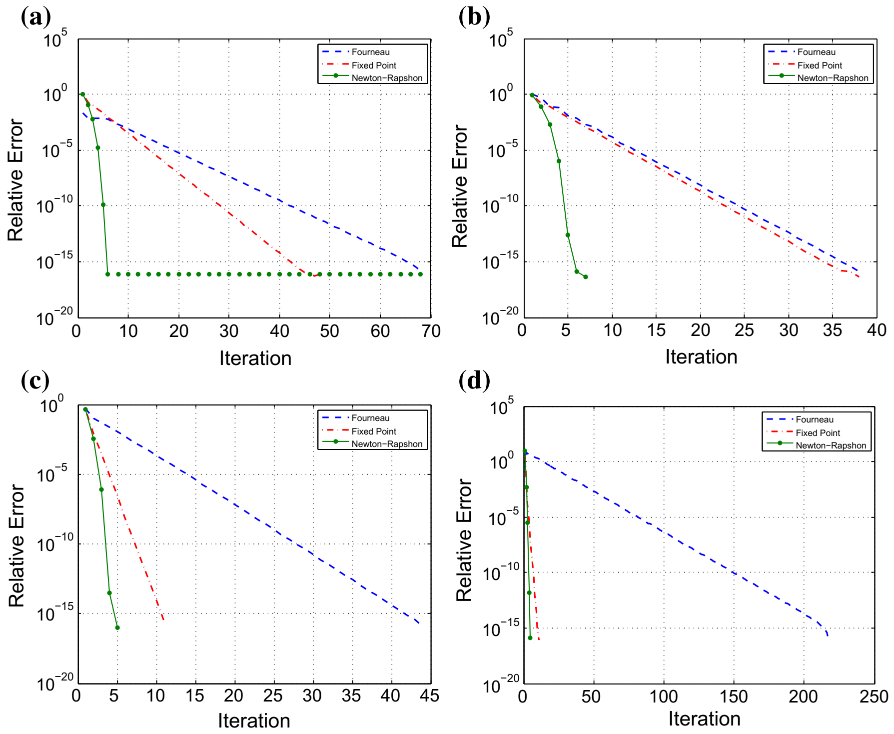

Fig. 2 Relative errors for the three methods. **a** $x = 0.5, \theta = 1, \gamma = 0.01$. **b** $x = 0.05, \theta = 1, \gamma = 0.01$. **c** $x = 0.95, \theta = 1, \gamma = 1$. **d** $x = 0.95, \theta = 100, \gamma = 1$

the larger the probability that a customers, after being served, leaves a queue as a positive customer rather than as a negative one. We set, for a given $\theta, \gamma \in \mathbb{R}_+$,

$$\mu = \theta \mathbf{1}, \quad \Lambda^+ = \mathbf{1}, \quad \Lambda^- = \gamma \mathbf{1}.$$

The external arrival rate of positive customers Λ^+ is kept fixed. We perform various experiments varying the weights of the convex combination for matrices P^+, P^- and the magnitude of the vectors μ, Λ^-. Varying γ we can control the amount of negative customers arriving in the network from the outside, while varying θ we control the service rate of the queues. The parameters x, θ, γ have been selected in such a way that the computed probabilities q_i are strictly less than one, ensuring the stability of the network.

6.1 Convergence

In Fig. 2 we have reported the relative errors $e_{FRN}^{(k)}, e_{FP}^{(k)}, e_{NR}^{(k)}$ in typical scenarios. We used $N = 10$ in all the experiments.

These plots confirm that *FRN* and *FP* have linear rate of convergence, while the Newton-Raphson iteration has quadratic convergence rate. In all cases *FP* converges faster than *FRN*, but the two methods perform differently: for small x (see Fig. 2b)

Table 1 Convex combination structure

θ	γ	*FRN*	*FP*	*NR*	$\max q_i$	$\rho(T'(z^*))$	R_{FRN}
$\mathbf{x = 0.05}$							
1	$\mathbf{10^{-2}}$	**38**	**36**	**6**	$\mathbf{7.39 \times 10^{-1}}$	$\mathbf{3.58 \times 10^{-1}}$	$\mathbf{3.7 \times 10^{-1}}$
1	1	22	20	5	4.49×10^{-1}	1.58×10^{-1}	1.73×10^{-1}
1	10^2	7	6	4	1×10^{-2}	8×10^{-5}	5×10^{-4}
10^2	10^{-2}	15	9	4	1.05×10^{-2}	9.2×10^{-3}	5.42×10^{-2}
10^2	1	15	9	4	1.1×10^{-2}	9×10^{-3}	5.4×10^{-2}
10^2	10^2	13	6	3	5×10^{-3}	2×10^{-3}	2×10^{-4}
$\mathbf{x = 0.5}$							
1	$\mathbf{10^{-2}}$	**68**	**45**	**6**	$\mathbf{9.94 \times 10^{-1}}$	$\mathbf{4.39 \times 10^{-1}}$	$\mathbf{6.11 \times 10^{-1}}$
1	1	30	19	5	5.65×10^{-1}	1.39×10^{-1}	3.11×10^{-1}
1	10^2	9	5	4	1×10^{-2}	4×10^{-5}	4×10^{-3}
10^2	10^{-2}	52	11	5	1.94×10^{-2}	1.4×10^{-2}	4.46×10^{-1}
10^2	1	50	10	5	1.91×10^{-2}	1.3×10^{-2}	4.44×10^{-1}
10^2	10^2	28	6	3	7×10^{-3}	2×10^{-3}	2.18×10^{-1}
$\mathbf{x = 0.95}$							
1	**1**	**46**	**12**	**5**	$\mathbf{9.35 \times 10^{-1}}$	$\mathbf{3.17 \times 10^{-2}}$	$\mathbf{4.53 \times 10^{-1}}$
1	10^2	12	5	4	1×10^{-2}	6×10^{-6}	9×10^{-3}
10^2	10^{-2}	207	10	5	7.22×10^{-2}	1.65×10^{-2}	8.38×10^{-1}
$\mathbf{10^2}$	**1**	**175**	**9**	**5**	$\mathbf{6.4 \times 10^{-2}}$	$\mathbf{1.12 \times 10^{-2}}$	$\mathbf{8.06 \times 10^{-1}}$
10^2	10^2	50	5	3	1×10^{-2}	4×10^{-4}	4.36×10^{-1}

Performance measures

FRN is particularly fast, reaching the same rate of *FP*, while for large x (see Fig. 2d) *FP* nearly reaches *NR* rate. In all cases *NR* requires only a few iterations to reach convergence, outperforming *FP* since both methods requires $O(N^3)$ time per step. In Sect. 6.2 a performance comparison between *NR* and *FRN* will be carried out: since *FRN* requires only $O(N^2)$ time per step, there will be a threshold $\overline{N}$ such that *NR* converges faster than *FRN* for $N \leq \overline{N}$. *FRN* requires a particular large number of iterations when x is close to one and the service rate μ is large, as we can see in Fig. 2d.

Table 1 reports, for $x = 0.05, 0.5, 0.95$, $\theta = 1, 100$ and $\gamma = 0.01, 1, 100$, the number of iterations necessary to reach a relative error of the order of 10^{-16}, the value $\max_{i=1,\dots,N} q_i$, the spectral radius $\rho(T'(z^*))$ of the Jacobian matrix of $T(z)$ in z^* and the convergence ratio of the Fourneau iteration

$$R_{FRN} = \lim_{k\to\infty} \frac{\|\overline{q}^{(k+1)} - \overline{q}^*\|_\infty}{\|\overline{q}^{(k)} - \overline{q}^*\|_\infty}$$

for $k \geq 0$. The convergence ratio of the fixed point iteration is not reported as in all cases it resulted equal to the upper bound $\rho(T'(z^*))$ (see Theorem 10) up to four decimal digits.

The rows in bold correspond to the plots in Fig. 2. We make the following remarks:

- The smaller the negative customers outside arrival rate γ, the higher the steady-state occupation probabilities q_i-s, and the smaller the positive customers traffic inside the network x the smaller the q_i-s;
- $\rho(T'(z^*))$ is decreasing in γ;
- The number of iteration of *FRN* is increasing in x, while the number of iterations of *FP* is decreasing in x;
- Increasing the service rate ($\theta = 100$) reduces the occupation probabilities q_i-s, as one would expect, and it has opposite effects on the performance of *FNR*: for small x (i.e. the internal traffic is mostly constituted by *negative* customers), it reduces the number of iterations, while for large x, it increases it.

We conclude that the smaller the value of Λ^-, the slower the convergence of *FP* and *FRN*. In particular, *FRN* performs better for small x, while *FP* performs better for large x. In any case, *FP* requires less iterations than *FRN*. Moreover, *FRN* performs particularly well for small x and large μ, and particularly bad for large x and large μ. Finally, *NR* requires sensibly less iterations than *FRN* and *FP* in all cases.

6.2 CPU time

In this section we will compare the CPU time requested by the Fourneau and the Newton-Raphson iteration for several values of the dimension N. We have shown that *NR* requires just a few iterations to reach convergence, while *FRN*, in some cases, can be very slow. This is due to the fact that *NR* has a quadratic rate of convergence, while *FRN* has a linear rate, where the reduction of the error at each step can be close to 1. However, *FRN* has a computational cost per step which grows as N^2, while *NR* has a cost per step which grow as N^3, therefore we expect that there will be a threshold $\overline{N}$ such that *NR* is preferable for the dimensions $N \leq \overline{N}$. This property will make *NR* a suitable algorithm for applications such as the ones in [5,14], where the need is to solve a huge number of different nonlinear equations (7) with small size N.

We do not perform the same comparison between *FRN* and *FP* as they are both linearly convergent methods, while *FP* has a greater cost per step than *FRN*, making it slower in almost all cases.

Below we report the results for the following case studies, where *FRN* performs differently.

- Case A: $x = 0.05$, $\mu = 10^2 \cdot \mathbf{1}$, $\Lambda^+ = \Lambda^- = \mathbf{1}$ (*FRN* performance is particularly good).
- Case B: $x = 0.5$, $\mu = 1 \cdot \mathbf{1}$, $\Lambda^+ = \mathbf{1}$, $\Lambda^- = 10^{-2} \cdot \mathbf{1}$ (*FRN* performance is average).
- Case C: $x = 0.95$, $\mu = 10^2 \cdot \mathbf{1}$, $\Lambda^+ = \Lambda^- = \mathbf{1}$ (*FRN* performance is particularly bad).

For each case and for each value of N, many simulations have been performed and the average CPU time over all the simulations is taken.

We observe that in Case A (Fig. 3a), the threshold is $\overline{N} = 14$ and for smaller values of N *NR* is slightly faster than *FRN*. In case B (Fig. 3b), the threshold is $\overline{N} = 44$:

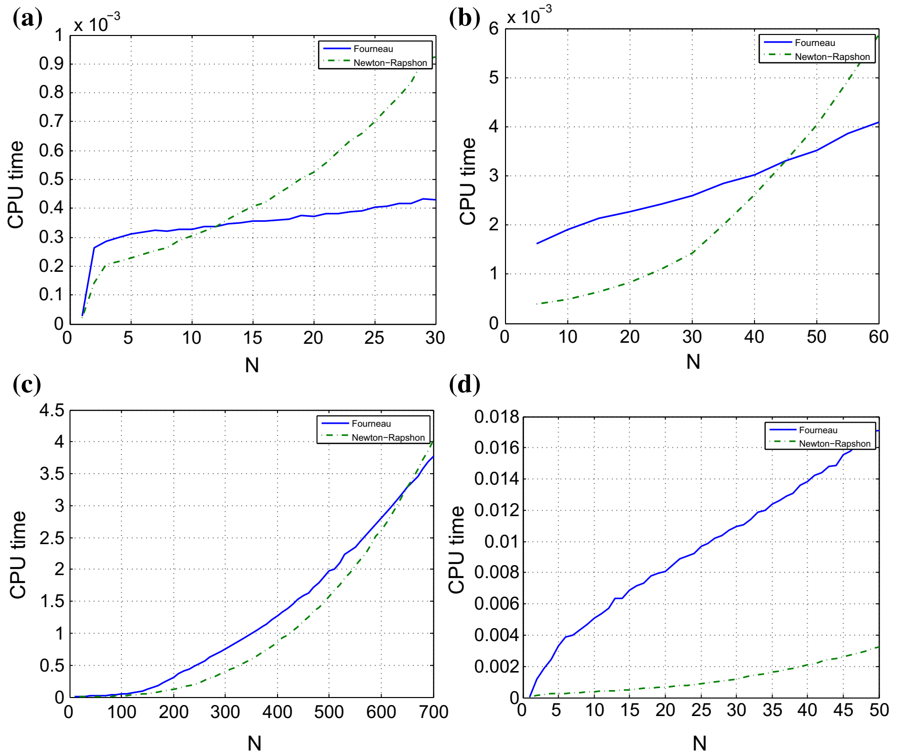

Fig. 3 CPU time comparison between *NR* and *FRN*. **a** Case A. **b** Case B. **c** Case C. **d** Case C, zoom

NR is faster than *FRN* for small values of N. For $N = 10$, the dimension used in the experiments in Sect. 6.1, *NR* is around 4 times faster than *FRN*. In case C (Fig. 3c), the threshold is around $\overline{N} = 620$: *FRN* is very slow in this case, making *NR* a preferable choice for moderate/large values of N. In particular, as reported in Fig. 3c, *NR* is around 13 times faster than *FRN* for $N = 10$, 7.5 times faster for $N = 40$, 3.5 times faster for $N = 70$ and roughly 2 times faster for larger values of N up to 300.

These results encourage the use of *NR* in problems where the computation of the steady-state distributions of a G-network must be carried out a large number of times, as in the context of supervised learning. This is because learning algorithms, as described in [18], requires at each step the computation of the steady-state distribution of a suitable G-network. In [18] it is advised to use the Fourneau algorithm, due to its $O(N^2)$ cost per step. However, as we have shown in this section, there are cases in which the Newton-Raphson method is several times faster than the Fourneau iteration, improving the performance of the learning algorithm. For instance, in [5] the authors tackle the Traffic Matrix Estimation problem by means of training several G-networks, each having dimension ranging from 9 to 14. They trained a total of $m = 132$ G-network using a learning dataset composet of 288 input-output pairs, and according to the gradient descent methods described in [9] this means that the system (7) must be solved a number of times proportional to $132 \times 288 = 38{,}016$. In [5] the authors used particularly simple networks with three-layer feed-forward structure, for which no particular algorithm is required in order to solve the correspondent systems (7). The results obtained in [5] suggest that the approach can be refined by employing fully recurrent G-networks, without restricting only to feed-forward structure. For such networks efficient algorithms for the solution of the system (7) are needed, and given the small dimension of the G-networks, employing the Newton-Raphson algorithm in this setting can substantially improve the performances.

Acknowledgements The authors wish to thank Dario Bini for the discussions on convergence properties and Stefano Giordano for providing introduction to the subject of random neural networks and for the interesting conversations on topics related to the paper. The authors wish also to thank the anonymous referees who helped to improve the presentation and pointed out some inaccuracies in the original version. Funding was provided by Università di Pisa (PRA 2015) and GNCS of INdAM.

References

1. Alfa, S.A., Xue, J., Ye, Q.: Entrywise perturbation theory for diagonally dominant M-matrices with applications. Numer. Math. **90**(3), 401–414 (2002)
2. Atalay, V., Gelenbe, E., Yalabik, N.: Image texture generation with the random neural network model. In: Kohonen, T. (ed) International Conference on Artificial Neural Networks (icann-91), Helsinki (1991)
3. Berman, A., Plemmons, R.J.: Nonnegative matrices in the mathematical sciences. In: Classics in applied mathematics, vol. 9. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (1994) (revised reprint of the 1979 original)
4. Bini, D.A., Latouche, G., Meini, B.: Numerical Methods for Structured Markov Chains. Oxford University Press, New York (2005)
5. Casas, P., Vaton, S.: On the use of random neural networks for traffic matrix estimation in large-scale IP networks. In: Proceedings of the 6th International Wireless Communications and Mobile Computing Conference, pp. 326–330. ACM (2010)
6. Fourneau, J.M.: Computing the steady-state distribution of networks with positive and negative customers. In: 13th IMACS World Congress on Computation and Applied Mathematics, Dublin (1991)
7. Gelenbe, E.: Random neural networks with negative and positive signals and product form solution. Neural Comput. **1**(4), 502–510 (1989)
8. Gelenbe, E.: Stability of the random neural network model. Neural Comput. **2**(2), 239–247 (1990)
9. Gelenbe, E.: Learning in the recurrent random neural network. Neural Comput. **5**(1), 154–164 (1993)
10. Gelenbe, E., Batty, F.: Minimum cost graph covering with the random neural network. Comput. Sci. Oper. Res. **1**, 139–147 (1992)
11. Hautphenne, S., Latouche, G., Remiche, M.A.: Newtons iteration for the extinction probability of a Markovian binary tree. Linear Algebra Appl. **428**(11), 2791–2804 (2008)
12. Istrc, V., et al.: Fixed Point Theory: An Introduction, vol. 1. Springer, Berlin (1981)
13. Latouche, G.: Newton's iteration for non-linear equations in Markov chains. IMA J. Numer. Anal. **14**(4), 583–598 (1994)
14. Mohamed, S., Rubino, G., Varela, M.: Performance evaluation of real-time speech through a packet network: a random neural networks-based approach. Perform. Eval. **57**(2), 141–161 (2004)
15. Ortega, J.M., Rheinboldt, W.C.: Iterative solution of nonlinear equations in several variables. In: Classics in applied mathematics, vol. 30. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2000) (reprint of the 1970 original)
16. Rubino, G., Tirilly, P., Varela, M.: Evaluating users satisfaction in packet networks using random neural networks. In: Artificial Neural Networks—ICANN 2006, pp. 303–312. Springer (2006)
17. Teke, A., Atalay, V.: Texture classification and retrieval using the random neural network model. Comput. Manag. Sci. **3**(3), 193–205 (2006)
18. Timotheou, S.: The random neural network: a survey. Comput. J. **53**(3), 251–267 (2010)
19. Varga, R.S.: Matrix iterative analysis. In: Computational Mathematics, vol. 27. Springer, Berlin (2000)