



Some Single Valued Neutrosophic Queueing Systems with Maple Code

Fatima Masri¹, Mohamed Bisher Zeina^{2*} and Omar Zeitouny³

¹ Department of Mathematical Statistics, Faculty of Science, University of Aleppo, Aleppo, Syria; fatenahmasri@gmail.com

² Department of Mathematical Statistics, Faculty of Science, University of Aleppo, Aleppo, Syria; bisher.zeina@gmail.com

³ Department of Mathematical Statistics, Faculty of Science, University of Aleppo, Aleppo, Syria; ozeitouny70@gmail.com

* Correspondence: bisher.zeina@gmail.com

Abstract:

In this paper we introduce for the first time the concept of single valued neutrosophic queueing systems (SVNQSs) as an extension of crisp and fuzzy queueing systems which are very applicable and important in controlling systems. SVNQSs have been defined assuming that arrival rates and departure rates are single valued neutrosophic trapezoidal numbers, and depending on this assumption probabilities and performance measures were also single valued neutrosophic trapezoidal numbers. Numerical examples were presented and solved successfully, and because of hard computations, a maple code is presented to make calculations easier.

Keywords: Single Valued Neutrosophic Set; Single Valued Trapezoidal Neutrosophic Number; Queueing Systems; Markovian Queues; Performance Measures

1. Introduction

Fuzzy sets presented by L.A. Zadeh assume that each element x belongs to a set $A \subseteq \Omega$ with membership degree $0 \leq \mu_A(x) \leq 1$ and doesn't belong to the set with non-membership degree $\mu_{A^c}(x)$ where $\mu_{A^c}(x) = 1 - \mu_A(x)$ [1]. This definition was extended by K. Atanassov by assuming that $0 \leq \mu_A(x) + \mu_{A^c}(x) \leq 1$ or $0 \leq \mu_A(x) + \nu_A(x) \leq 1$ [2]. Recently F. Smarandache presented what is known by neutrosophic set as a generalization to fuzzy sets and intuitionistic fuzzy sets assuming that $0 \leq \mu_A(x) + \delta_A(x) + \nu_A(x) \leq 3$ where $\delta_A(x)$ is degree of indeterminacy, also F. Smarandache supposed that all these three components are subsets of nonstandard real intervals $]^{-}0, 1^{+}[$. When these components are taken as subsets of standard real intervals, we have what is named by single valued neutrosophic set, i.e., when $\mu_A(x), \delta_A(x), \nu_A(x) \in [0,1]$ and $0 \leq \mu_A(x) + \delta_A(x) + \nu_A(x) \leq 3$ [3] [4].

Due to this generalization, many extensions to all branches of science have been made including probability theory, statistics, reliability theory, queueing theory, artificial intelligence, data mining, algebra, linear algebra, mathematical analysis, complex analysis, differential equations, physics, philosophy etc [5] [6] [7] [8] [9] [10] [11] [12] [13] [14].

In probability theory, neutrosophic probability measure was defined by F.Smarandache as a mapping $NP: X \rightarrow [0,1]^3$ where X is a neutrosophic sample space and the neutrosophic probability function of an event $A \subseteq \Omega$ is defined by $NP(A) = (ch(A), ch(neutA), ch(antiA)) = (\alpha, \beta, \gamma)$ where $0 \leq \alpha, \beta, \gamma \leq 1$ and $0 \leq \alpha + \beta + \gamma \leq 3$, i.e., $NP(A)$ can be called a single valued probability measure.

researchers introduced many neutrosophic probability distributions like Poisson, exponential, binomial, normal, uniform, Weibull, ...etc.

Later, M.B. Zeina and A. Hatip in [15] defined the neutrosophic random variable in the form $X_N = X + I$ where I is indeterminacy which satisfies $I^2 = I$ and then defined many probabilistic properties based on the last definition, which opens new philosophy to the study probability measure and many new definitions and extensions were made depending on it like in [16] [17].

One of the most important applications of probability theory is queueing theory presented by A.K. Erlang when he created mathematical models describing systems of telephone exchanges and then this theory has been applied in many telecommunication systems [18].

Queueing theory was generalized to fuzzy queueing theory to fit uncertainty in telecommunication systems in many papers like [19] [20] [21]. Many neutrosophic queueing systems were defined and studied by M.B. Zeina in many papers like [22] [23] [24] [25].

In the previous studies about neutrosophic queueing systems only indeterminacy was taken in hand where systems have been studied assuming that parameters are of the form $a + bI$ where I is indeterminacy. In this paper we will study neutrosophic queueing systems assuming that parameters (Arrivals and Departures) are single valued trapezoidal neutrosophic numbers as a more general case of fuzzy and intuitionistic fuzzy queueing systems, then we present a maple code that makes calculations easier.

2. Preliminaries

2.1 Neutrosophic Set [4]

Let E be a universe. Then, a neutrosophic set A over E is defined by:

$$A = \{ \langle x, (T_A(x), I_A(x), F_A(x)) \rangle ; x \in E \}$$

Where:

$T_A : E \rightarrow]^{-0}, 1^{+}[$ is called truth-membership function.

$I_A : E \rightarrow]^{-0}, 1^{+}[$ indeterminacy-membership function.

$F_A : E \rightarrow]^{-0}, 1^{+}[$ falsity-membership function.

$$^{-0} \leq T_A(x) + I_A(x) + F_A(x) \leq 3^{+}$$

2.2 Single Valued Neutrosophic Set [4]

Let E be a universe. Then, a single valued neutrosophic set A over E is defined by:

$$A = \{ \langle x, (T_A(x), I_A(x), F_A(x)) \rangle ; x \in E \}$$

Where:

$T_A : E \rightarrow [0,1]$ is called truth-membership function.

$I_A : E \rightarrow [0,1]$ indeterminacy-membership function.

$F_A : E \rightarrow [0,1]$ falsity-membership function.

$$0 \leq T_A(x) + I_A(x) + F_A(x) \leq 3$$

This set is used in engineering, biology, decision-making and other applications because its concept is clearer and more applied than neutrosophic sets.

2.3 Single Valued Trapezoidal Neutrosophic Number (SVTNN) [26]

A single valued trapezoidal neutrosophic number is denoted by:

$$\tilde{a} = \langle (a_1, b_1, c_1, d_1) ; w_{\tilde{a}}, u_{\tilde{a}}, y_{\tilde{a}} \rangle$$

Where:

$w_{\tilde{a}}$ is the nucleus of truth membership, $u_{\tilde{a}}$ is the nucleus of indeterminacy membership and $y_{\tilde{a}}$ is the nucleus of falsity membership where:

Truth-membership function is:

$$T_{\tilde{a}}(x) = \begin{cases} \frac{(x - a_1)w_{\tilde{a}}}{b_1 - a_1} & ; a_1 \leq x < b_1 \\ w_{\tilde{a}} & ; b_1 < x \leq c_1 \\ \frac{(d_1 - x)w_{\tilde{a}}}{d_1 - c_1} & ; c_1 < x \leq d_1 \\ 0 & ; \text{otherwise} \end{cases}$$

Indeterminacy-membership function is:

$$I_{\tilde{a}}(x) = \begin{cases} \frac{(b_1 - x + u_{\tilde{a}}(x - a_1))}{b_1 - a_1} & ; a_1 \leq x < b_1 \\ u_{\tilde{a}} & ; b_1 < x \leq c_1 \\ \frac{(x - c_1 + u_{\tilde{a}}(d_1 - x))}{d_1 - c_1} & ; c_1 < x \leq d_1 \\ 1 & ; \text{otherwise} \end{cases}$$

Falsity-membership function is:

$$F_{\tilde{a}}(x) = \begin{cases} \frac{(b_1 - x + y_{\tilde{a}}(x - a_1))}{b_1 - a_1} & ; a_1 \leq x < b_1 \\ y_{\tilde{a}} & ; b_1 < x \leq c_1 \\ \frac{(x - c_1 + y_{\tilde{a}}(d_1 - x))}{d_1 - c_1} & ; c_1 < x \leq d_1 \\ 1 & ; \text{otherwise} \end{cases}$$

2.3 Operations on SVTNNs [26]

Let $\tilde{a} = \langle (a_1, b_1, c_1, d_1) ; w_{\tilde{a}}, u_{\tilde{a}}, y_{\tilde{a}} \rangle, \tilde{b} = \langle (a_2, b_2, c_2, d_2) ; w_{\tilde{b}}, u_{\tilde{b}}, y_{\tilde{b}} \rangle$ be two SVTNNs and $\gamma \geq 0$ any real number, then:

$$\begin{aligned} \tilde{a} \oplus \tilde{b} &= \langle (a_1 + a_2, b_1 + b_2, c_1 + c_2, d_1 + d_2) ; w_{\tilde{a}} + w_{\tilde{b}} - w_{\tilde{a}}w_{\tilde{b}}, u_{\tilde{a}}u_{\tilde{b}}, y_{\tilde{a}}y_{\tilde{b}} \rangle \\ \tilde{a} \otimes \tilde{b} &= \langle (a_1a_2, b_1b_2, c_1c_2, d_1d_2) ; w_{\tilde{a}}w_{\tilde{b}}, u_{\tilde{a}} + u_{\tilde{b}} - u_{\tilde{a}}u_{\tilde{b}}, y_{\tilde{a}} + y_{\tilde{b}} - y_{\tilde{a}}y_{\tilde{b}} \rangle \\ \gamma \tilde{a} &= \langle (\gamma a_1, \gamma b_1, \gamma c_1, \gamma d_1) ; 1 - (1 - w_{\tilde{a}})^\gamma, u_{\tilde{a}}^\gamma, y_{\tilde{a}}^\gamma \rangle \\ \tilde{a}^\gamma &= \langle (a_1^\gamma, b_1^\gamma, c_1^\gamma, d_1^\gamma) ; w_{\tilde{a}}^\gamma, 1 - (1 - u_{\tilde{a}})^\gamma, 1 - (1 - y_{\tilde{a}})^\gamma \rangle \end{aligned}$$

3. Recall of Some Crisp Queueing Systems [18]

Here we recall definitions and properties of some crisp queueing systems including M/M/1, M/M/c, M/M/c/b.

3.1 M/M/1 Queueing System

In this system we have one server with arrival rate λ customers per time unit and serving rate η customers per time unit.

The probability that we will not find customers in the system is:

$$P_0 = (1 - \rho) \quad ; \quad \rho = \frac{\lambda}{\eta}$$

The probability that we will find n customers in the system is:

$$P_n = (1 - \rho) * \rho^n \quad ; \quad n = 0, 1, 2, \dots$$

The average number of customers in queue is:

$$L_q = \frac{\rho^2}{1 - \rho}$$

The average number of customers in system is:

$$L_s = \frac{\rho}{1 - \rho}$$

The mean waiting time in queue is:

$$W_q = \frac{L_q}{\lambda}$$

The mean waiting time in system is:

$$W_s = \frac{L_s}{\lambda}$$

3.2 M/M/c Queueing System

In this system we have c servers with arrival rate λ and serving rate η .

The probability that we will not find customers in the system is:

$$P_0 = \frac{1}{\sum_{n=0}^{c-1} \frac{\rho^n}{n!} + \frac{\rho^c}{c! * (1 - \frac{\rho}{c})}} \quad ; \quad \rho = \frac{\lambda}{\eta}, \quad \frac{\rho}{c} < 1$$

The probability that we will find n customers in the system is:

$$P_n = \begin{cases} \frac{\rho^n}{n!} * P_0 & ; n = 0, 1, \dots, c - 1 \\ \frac{\rho^n}{c!} * \frac{P_0}{c^{n-c}} & ; n = c, c + 1, \dots \end{cases}$$

The average number of customers in queue is:

$$L_q = \frac{\rho^{c+1}}{c!} * \frac{P_0}{c * (1 - \frac{\rho}{c})^2}$$

The average number of customers in system is:

$$L_s = L_q + \rho$$

The mean waiting time in queue is:

$$W_q = \frac{L_q}{\lambda}$$

The mean waiting time in system is:

$$W_s = \frac{L_s}{\lambda}$$

3.3 M/M/c/b Queueing System

Here we have c servers and the system size is limited by b where $c < b$ with arrival rate λ and serving rate η .

The probability that we will not find customers in the system is:

$$P_0 = \frac{1}{\sum_{n=0}^{c-1} \frac{\rho^n}{c!} + \frac{\rho^c}{c!} * \left(\frac{1 - (\frac{\rho}{c})^{b-c+1}}{1 - \frac{\rho}{c}} \right)} ; \rho = \frac{\lambda}{\eta}$$

The probability that we will find n customers in the system is:

$$P_n = \begin{cases} \frac{\rho^n}{n!} * P_0 & ; n = 0, 1, \dots, c - 1 \\ \frac{\rho^n}{c!} * \frac{P_0}{c^{n-c}} & ; n = c, c + 1, \dots, b \end{cases}$$

The average number of customers in queue is:

$$L_q = \frac{\rho^c * P_0}{c!} * \sum_{n=c+1}^b (n - c) * \left(\frac{\rho}{c}\right)^{n-c}$$

The effective arrival rate is:

$$\lambda_e = \lambda * (1 - P_b)$$

The average number of customers in system is:

$$L_s = L_q + \frac{\lambda_e}{\eta}$$

The mean waiting time in queue is:

$$W_q = \frac{L_q}{\lambda_e}$$

The mean waiting time in system is:

$$W_s = \frac{L_s}{\lambda_e}$$

4. Single Valued Neutrosophic Queueing Systems

Here we suppose that both of arrival rate and serving rate are SVTNNs given by

$$N\lambda = \langle (\lambda_1, \lambda_2, \lambda_3, \lambda_4); w_\lambda, u_\lambda, v_\lambda \rangle, N\eta = \langle (\eta_1, \eta_2, \eta_3, \eta_4); w_\eta, u_\eta, v_\eta \rangle$$

Where:

$$0 \leq w_\lambda, u_\lambda, v_\lambda \leq 1 \quad \& \quad 0 \leq w_\eta, u_\eta, v_\eta \leq 1$$

$$0 \leq w_\lambda + u_\lambda + v_\lambda \leq 3 \quad \& \quad 0 \leq w_\eta + u_\eta + v_\eta \leq 3$$

The neutrosophic utilization coefficient is:

$$N\rho = \frac{N\lambda}{N\eta} = \left\langle \left(\frac{\lambda_1}{\eta_4}, \frac{\lambda_2}{\eta_3}, \frac{\lambda_3}{\eta_2}, \frac{\lambda_4}{\eta_1} \right); w_\lambda * w_\eta, u_\lambda + u_\eta - u_\lambda * u_\eta, v_\lambda + v_\eta - v_\lambda * v_\eta \right\rangle$$

We will depend on all of our results on neutrosophic operations defined in 2.3

4.1 Single Valued Neutrosophic M/M/1 Queueing System

In this system we have one server and infinite queue size and based on (2.3,3.1) we can find the following:

Neutrosophic probability that we will not find customers in the system

$$NP_0 = \left\langle (1 - \rho_4, 1 - \rho_3, 1 - \rho_2, 1 - \rho_1); w_\rho, u_\rho, v_\rho \right\rangle \quad (1)$$

Where w_ρ, u_ρ, v_ρ are truth, indeterminacy and falsity of neutrosophic utilization coefficient respectively.

The neutrosophic probability that we will find n customers in the system will be:

$$NP_n = \left\langle ((1 - \rho_4)\rho_1^n, (1 - \rho_3)\rho_2^n, (1 - \rho_2)\rho_3^n, (1 - \rho_1)\rho_4^n); w_\rho * w_\rho^n, 1 - (1 - u_\rho)^n - (1 - (1 - u_\rho)^n)u_\rho + u_\rho, 1 - (1 - v_\rho)^n - (1 - (1 - v_\rho)^n)v_\rho + v_\rho \right\rangle ;$$

$$n = 0, 1, 2, \dots \quad (2)$$

The neutrosophic average number of customers in queue is:

$$NL_q = \left\langle \left(\frac{\rho_1^2}{1 - \rho_4}, \frac{\rho_2^2}{1 - \rho_3}, \frac{\rho_3^2}{1 - \rho_2}, \frac{\rho_4^2}{1 - \rho_1} \right); w_\rho^3, 1 - (1 - u_\rho)^2 - (1 - (1 - u_\rho)^2)u_\rho + u_\rho, 1 - (1 - v_\rho)^2 - (1 - (1 - v_\rho)^2)v_\rho + v_\rho \right\rangle \quad (3)$$

The neutrosophic average number of customers in system is:

$$NL_s = \left\langle \left(\frac{\rho_1}{1 - \rho_4}, \frac{\rho_2}{1 - \rho_3}, \frac{\rho_3}{1 - \rho_2}, \frac{\rho_4}{1 - \rho_1} \right); w_\rho^2, 2u_\rho - u_\rho^2, 2v_\rho - v_\rho^2 \right\rangle \quad (4)$$

The neutrosophic mean waiting time in queue is:

$$NW_q = \frac{NL_q}{N\lambda} = \left\langle \left(\frac{\rho_1^2}{\lambda_4(1 - \rho_4)}, \frac{\rho_2^2}{\lambda_3(1 - \rho_3)}, \frac{\rho_3^2}{\lambda_2(1 - \rho_2)}, \frac{\rho_4^2}{\lambda_1(1 - \rho_1)} \right); w_\lambda * w_\rho^3, 1 - (1 - u_\rho)^2 - (1 - (1 - u_\rho)^2)u_\rho + u_\rho + u_\lambda - (1 - (1 - u_\rho)^2 - (1 - (1 - u_\rho)^2)u_\rho + u_\rho) * u_\lambda, 1 - (1 - v_\rho)^2 - (1 - (1 - v_\rho)^2)v_\rho + v_\rho + v_\lambda - (1 - (1 - v_\rho)^2 - (1 - (1 - v_\rho)^2)v_\rho + v_\rho) * v_\lambda \right\rangle \quad (5)$$

The neutrosophic mean waiting time in system is:

$$NW_s = \frac{NL_s}{N\lambda} = \left\langle \left(\frac{\rho_1}{\lambda_4(1 - \rho_4)}, \frac{\rho_2}{\lambda_3(1 - \rho_3)}, \frac{\rho_3}{\lambda_2(1 - \rho_2)}, \frac{\rho_4}{\lambda_1(1 - \rho_1)} \right); w_\rho^3, 2u_\rho - u_\rho^2 + u_\lambda - (2u_\rho - u_\rho^2) * u_\lambda, 2v_\rho - v_\rho^2 + v_\lambda - (2v_\rho - v_\rho^2) * v_\lambda \right\rangle \quad (6)$$

Example 1 (Single Valued M/M/1 Queue)

Suppose that both arrival rate and serving rate are SVTNNs given by

$$N\lambda = \langle (1, 2, 3, 4); 0.8, 0.2, 0.01 \rangle, N\eta = \langle (5, 6, 7, 8); 0.9, 0.1, 0.01 \rangle$$

The neutrosophic probability that we will find no customers in systems using equation (1) will be:

$$NP_0 = \langle (0.2000000000, 0.5000000000, 0.7142857143, 0.8750000000), 0.72, 0.28, 0.0199 \rangle$$

Which means that with possibility 72% the probability of finding 0 customers in the system will range between 50% and 71.43% and will never be less than 20% or more than 87.5%.

We are also unsure with percentage of indeterminacy 28% of these results and we may be false by 1.9% falsity degree.

Fig1. Below shows all the possibilities of different truth, indeterminacy and falsity degrees by drawing a horizontal line on the graph, e.g. if we draw a horizontal line at y=0.4 we say that: with truth degree 40% P0 will lay between about 0.36 and about 0.78, with indeterminacy degree 60% (which is 1-y) P0 will lay between 0.45 and 0.74 and with falsity degree 60% (which is also 1-y) P0 will lay between 0.44 and 0.75.

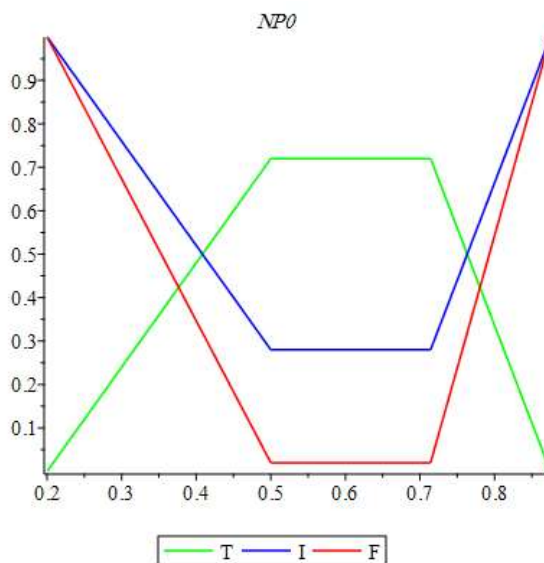


Fig1. NP0 in single valued neutrosophic M/M/1 queue

The neutrosophic probability that we will find 3 customers in system using equation (2) will be:

$$NP_3 = < (0.0003906250000, 0.01166180758, 0.08928571429, 0.4480000000), \\ 0.26873856, 0.73126144, 0.07725530557 >$$

Which means that with possibility 26.9% the probability of finding 3 customers in the system will range between 1.2% and 8.9% and will never be less than 0.04% or more than 44.8%.

We are also unsure with percentage of indeterminacy 73% of these results and we may be false by 7.7% falsity degree.

Fig2. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NP3:

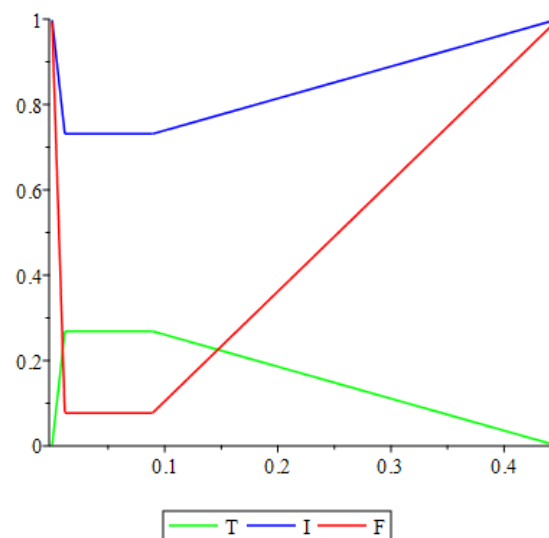


Fig2. NP3 in single valued neutrosophic M/M/1 queue

Neutrosophic performance measures using equations (3) to (6) will be:

$$NL_q = < (0.01785714286, 0.1142857143, 0.5000000000, 3.200000000), \\ 0.373248, 0.626752, 0.05851985060 >$$

Which means that surely 37% the average number of customers in queue will range between 11% and 50% and will never be less than 1.8% or more than 3.20.

We are also unsure with percentage of indeterminacy 62.7% of these results and we may be false by 5.8% falsity degree.

Fig3. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NLq:

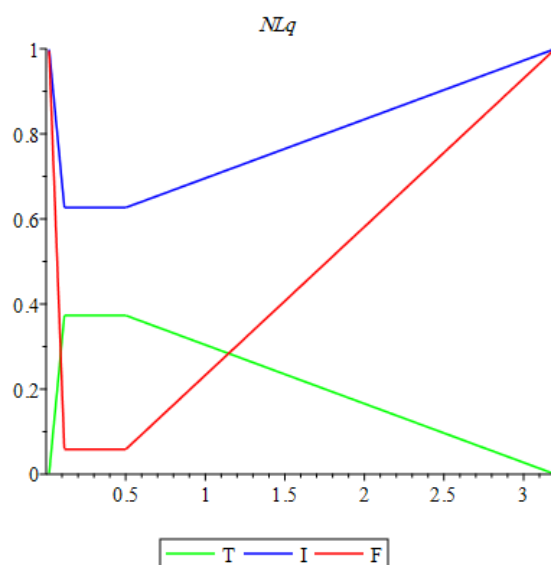


Fig3. NLq in single valued neutrosophic M/M/1 queue

$$NL_s = < (0.1428571429, 0.4000000000, 1., 4.), 0.5184, 0.4816, 0.03940399 >$$

Which means that surely 51.8% the average number of customers in system will range between 40% and 100% and will never be less than 14.3% or more than 4.

We are also unsure with percentage of indeterminacy 48% of these results and we may be false by 3.9% falsity degree.

Fig4. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NLs:

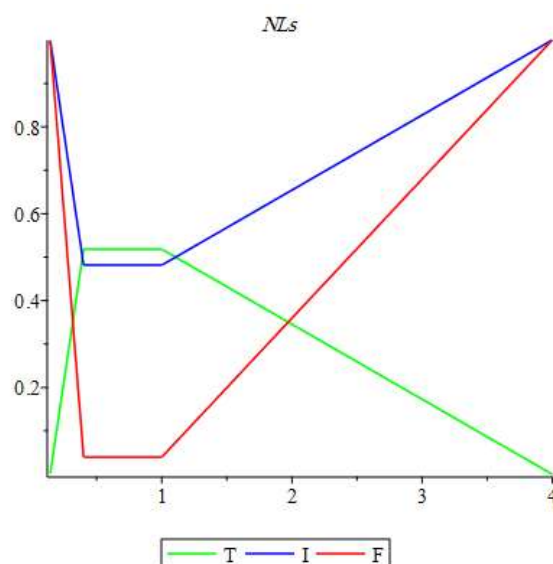


Fig4. NLs in single valued neutrosophic M/M/1 queue

$$NW_q = < (0.004464285715, 0.03809523810, 0.2500000000, 3.200000000), 0.2985984, 0.7014016, 0.06793465209 >$$

Which means that surely 29.8% the mean waiting time in queue will range between 3.8% and 25% and will never be less than 0.44% or more than 3.20.

We are also unsure with percentage of indeterminacy 70% of these results and we may be false by 6.8% falsity degree.

Fig5. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NWq:

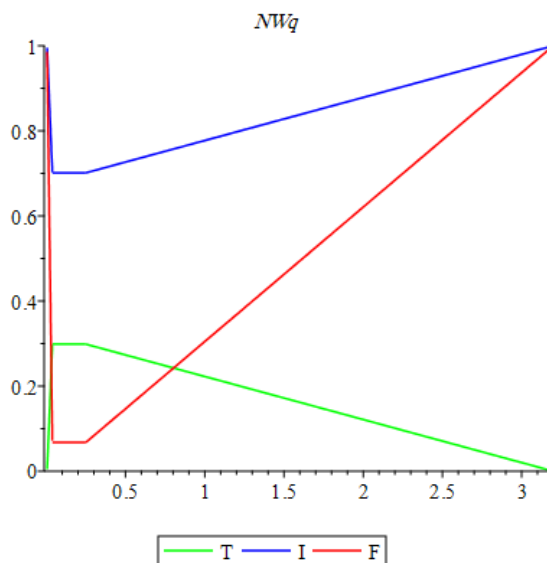


Fig5. NWq in single valued neutrosophic M/M/1 queue

$$NW_s = \langle (0.03571428572, 0.1333333333, 0.5000000000, 4.), \\ 0.41472, 0.58528, 0.0490099501 \rangle$$

Which means that surely 41% the mean waiting time in system will range between 13.3% and 50% and will never be less than 3.5% or more than 4.

We are also unsure with percentage of indeterminacy 58.5% of these results and we may be false by 4.9% falsity degree.

Fig6. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NWs:

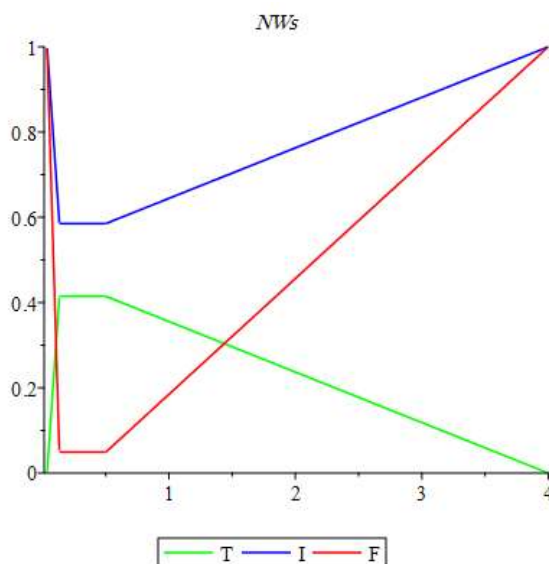


Fig6. NWs in single valued neutrosophic M/M/1 queue

4.2 Single Valued Neutrosophic M/M/c and M/M/c/b Queueing System

In M/M/c system we have c parallel servers working homogeneously with neutrosophic arrivals and neutrosophic departures and unlimited queue size. M/M/c/b is as same as M/M/c except that the first one has limited queue size. Finding probabilities of these queueing systems cannot be done by hand because of complex calculations so we will present a Maple package of code which makes calculations easier.

4.3 Single Valued Neutrosophic Queueing Systems Package

Notice that equations (1-6) are very hard to be applied by hand calculating, so we wrote a maple code depending on our previous neutrosophic package in [14] to handle neutrosophic queues easily.

A general procedure with overloading in parameters named QueueingSystem(NLambda,NMu,n,c,b) calls a procedure to calculate neutrosophic probability of finding 0 customers in the system, neutrosophic probability of finding n customers in the system, neutrosophic average number of customers in the system, neutrosophic average number of customers in the queue, neutrosophic mean time in the system and neutrosophic mean time in the queue then plots these numbers, where:

NLambda is $N\lambda$ (neutrosophic arrival rate)

NMu is $N\mu$ (neutrosophic departure rate)

n is number of customers that we want to calculate the neutrosophic probability of finding them in the system.

c is number of servers working in parallel.

b is size of system (including servers).

We can use this procedure to describe different types of queues as shown in table 1:

Procedure	Neutrosophic Queueing System
QueueingSystem(NLambda,NMu,n,c,b)	NM/NM/c/b
QueueingSystem(NLambda,NMu,n,c)	NM/NM/c/∞ or NM/NM/c
QueueingSystem(NLambda,NMu,n)	NM/NM/1/∞ or NM/NM/1

Table 1. Single valued neutrosophic queueing systems overloaded procedures.

Also, a plotting procedure is programmed to plot the results as follows:

SVTNQPlot(QueueingSystem(NLambda,NMu,OVERLOADED PARAMETERS),measure);

Where 'measure' can be replaced by p0,pn,lq,ls,wq or ws.

Example 2 (Single Valued M/M/c Queue)

suppose that we have a queueing system with arrival rate given as follows:

$$N\lambda = \langle (2,2.5,3,3.5); 0.8,0.2,0.01 \rangle, N\eta = \langle (5,5.5,6,6.5); 0.9,0.1,0.01 \rangle$$

We can define these rates using the code:

NLambda :=SVTN(2,2.5,3,3.5,0.8,0.2,0.01);NMu:=SVTN(5,5.5,6,6.5,0.9,0.1,0.01);

Suppose that we have 2 servers, and we would like to calculate the following:

1. Probability of finding no customers in the system.
2. Probability of finding 3 customers in the system.
3. Performance measures.

These can be done by typing:

QueueingSystem(NLambda,NMu,3,2);

Which results:

The neutrosophic probability that we will find no customers in systems using equation (1) will be:

$$NP_0 = \left[\begin{matrix} [0.4814814815, 0.5714285711, 0.6551724136, 0.7333333331] , \\ 0.96224256, 0.03775744, 0.00001560437408 \end{matrix} \right]$$

Which means that with possibility 96.2% the probability of finding 0 customers in the system will range between 57% and 65.5% and will never be less than 48% or more than 73.3%.

We are also unsure with percentage of indeterminacy 3.8% of these results and we may be false by 0.002% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),p0);

Fig7. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NP0:

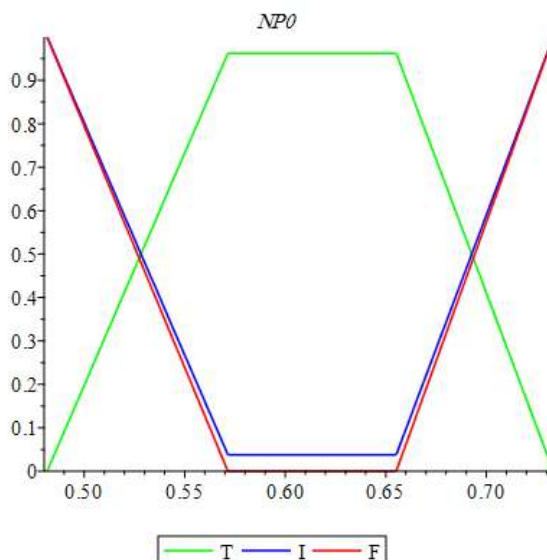


Fig7. NP0 in single valued neutrosophic M/M/c queue

The neutrosophic probability that we will find 3 customers in system using equation (2) will be:

$$NP_3 = \left[\begin{matrix} [0.003506465046, 0.01033399471, 0.02658099950, 0.0628833332] , \\ 0.1678434279, 0.8321565721, 0.2449034052 \end{matrix} \right]$$

Which means that with possibility 16.8% the probability of finding 3 customers in the system will range between 1.04% and 2.7% and will never be less than 0.35% or more than 6.3%.

We are also unsure with percentage of indeterminacy 83% of these results and we may be false by 24.5% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),pn);

Fig8. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NP3:

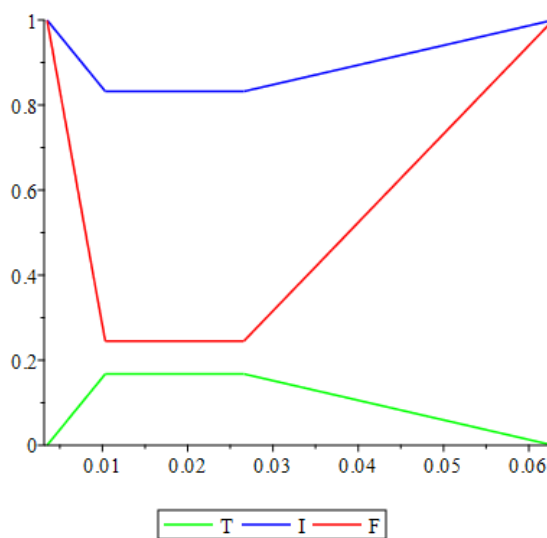


Fig8. NP3 in single valued neutrosophic M/M/c queue

Neutrosophic performance measures using equations (3) to (6) will be:

$$NL_q = \left[\begin{matrix} [0.004897459442, 0.01648858989, 0.05025470221, 0.1488362919] , \\ 0.2004575439, 0.7995424561, 0.2419205946 \end{matrix} \right]$$

Which means that surely 20% the average number of customers in queue will range between 1.65% and 5.03% and will never be less than 0.48% or more than 14.9%.

We are also unsure with percentage of indeterminacy 79.9% of these results and we may be false by 24.2% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),lq);

Fig9. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NLq:

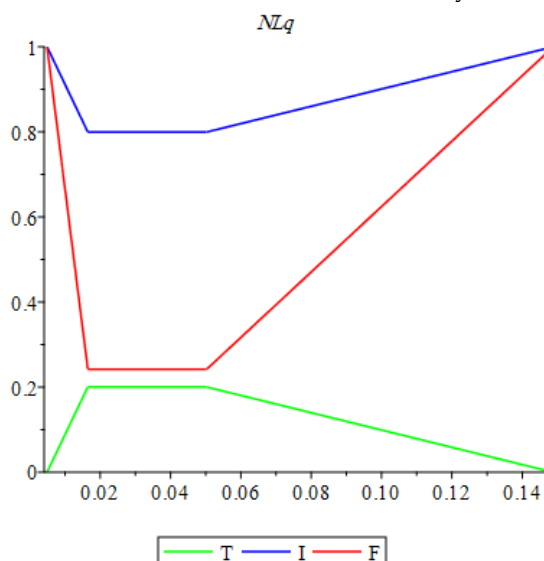


Fig9. NLq in single valued neutrosophic M/M/c queue

$$NL_s = \left[\begin{matrix} [0.3125897671, 0.4331552566, 0.5957092477, 0.8488362919], \\ 0.7761281123, 0.2238718877, 0.004814219833 \end{matrix} \right]$$

Which means that surely 77.6% the average number of customers in system will range between 43.3% and 59.6% and will never be less than 31.3% or more than 84.9%.

We are also unsure with percentage of indeterminacy 22.4% of these results and we may be false by 0.48% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),ls);

Fig10. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NLs:

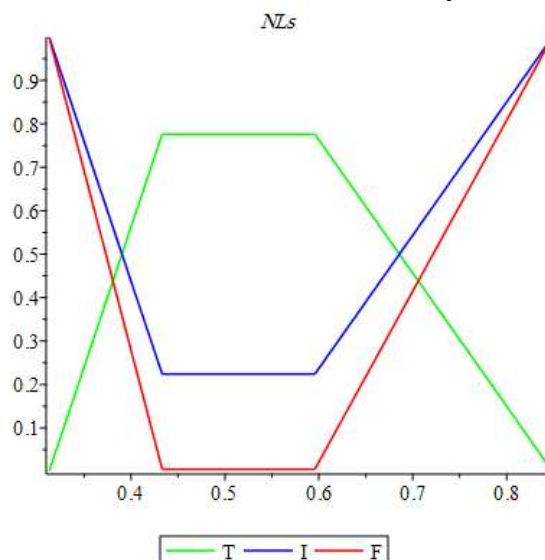


Fig10. NLs in single valued neutrosophic M/M/c queue

$$NW_q = \left[\begin{matrix} [0.001399274126, 0.005496196630, 0.02010188088, 0.07441814595e], \\ 0.1603660351, 0.8396339649, 0.2495013887 \end{matrix} \right]$$

Which means that surely 16% the mean waiting time in queue will range between 0.55% and 2.01% and will never be less than 0.14% or more than 7.44%.

We are also unsure with percentage of indeterminacy 83.9% of these results and we may be false by 24.9% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),wq);
 Fig11. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NWq:

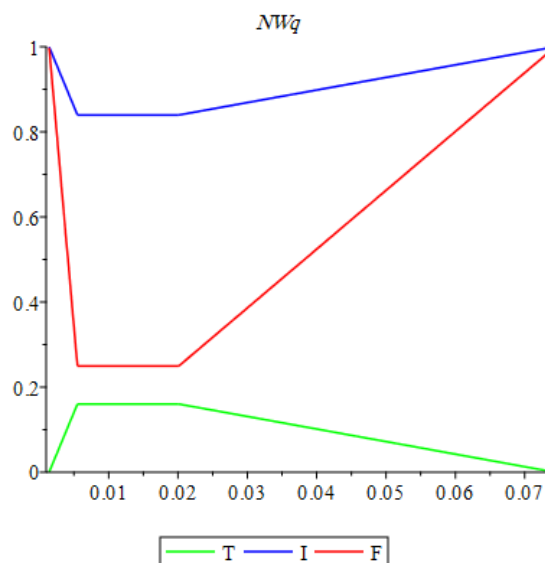


Fig11. NWq in single valued neutrosophic M/M/c queue

$$NW_s = \left[\begin{matrix} [0.08931136203, 0.1443850855, 0.2382836991, 0.4244181460] , \\ 0.6209024898, 0.3790975102, 0.01476607764 \end{matrix} \right]$$

Which means that surely 62% the mean waiting time in system will range between 14.4% and 24% and will never be less than 8.9% or more than 42%.

We are also unsure with percentage of indeterminacy 37.9% of these results and we may be false by 1.4% falsity degree.

SVTNQPlot(QueueingSystem(NLambda,NMu,3,2),ws);
 Fig12. Below shows all the possibilities of different truth, indeterminacy and falsity degrees of NWs:

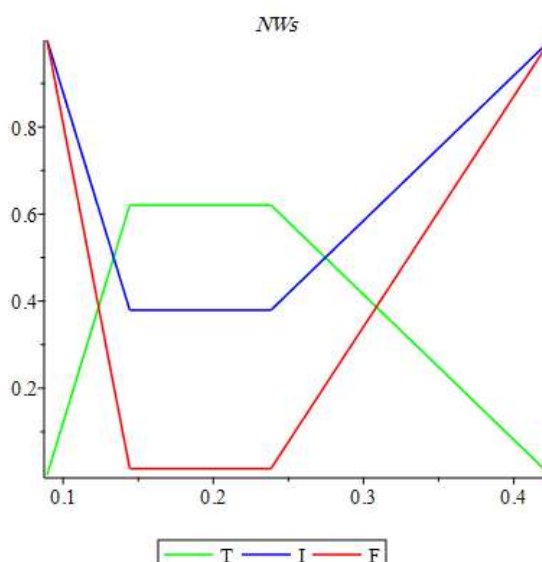


Fig12. NWs in single valued neutrosophic M/M/c queue

Example 3 (Single Valued M/M/c/b Queue)

Suppose that we have a queueing system with:

$$N\lambda = \langle (2,2.5,3,3.5); 1,0.01,0.01 \rangle, N\eta = \langle (4,4.5,5,5.5); 0.9,0.1,0.01 \rangle$$

Suppose that we have 2 servers and system size limited by 4, and we would like to calculate the following:

1. Probability of finding no customers in the system.
2. Probability of finding 4 customers in the system.
3. Performance measures.

These can be done by typing:

```
NLambda :=SVTN(2,2.5,3,3.5,1,0.01,0.01);NMu:=SVTN(4,4.5,5,5.5,0.9,0.1,0.01);
QueueingSystem(NLambda,NMu,4,2,4);
```

Which results:

The neutrosophic probability that we will find no customers in systems using equation (1) will be:

$$NP_0 = \left[\begin{array}{l} [0.3919316843, 0.5013054829, 0.6022304833, 0.6955662590] , \\ 0.9912331324, 0.01057498123, 0.0002232497869 \end{array} \right]$$

Which means that with possibility 99% the probability of finding 0 customers in the system will range between 50% and 60% and will never be less than 39% or more than 69.5%.

We are also unsure with percentage of indeterminacy 1.05% of these results and we may be false by 0.02% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),p0);
```

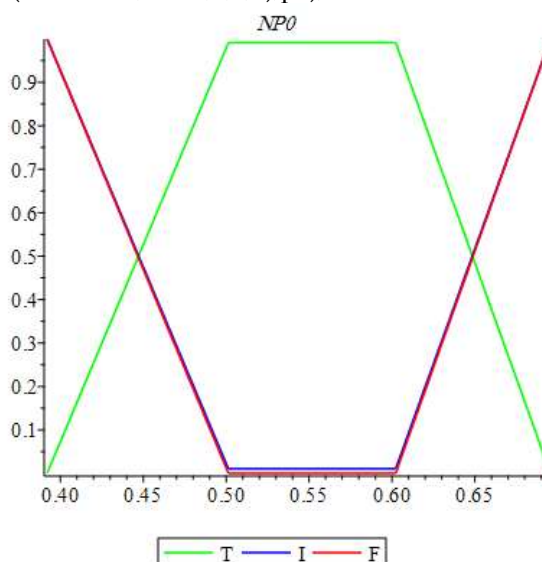


Fig13. NP0 in single valued neutrosophic M/M/c/b queue

The neutrosophic probability that we will find 4 customers in system using equation (2) will be:

$$NP_n = \left[\begin{array}{l} [0.0008566227644, 0.003916449084, 0.01486988847, 0.05096602137] , \\ 0.2870206713, 0.7337556823, 0.3662088099 \end{array} \right]$$

Which means that with possibility 28.7% the probability of finding 4 customers in the system will range between 0.39% and 1.5% and will never be less than 0.08% or more than 5.09%.

We are also unsure with percentage of indeterminacy 73.4% of these results and we may be false by 36.6% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),pn);
```

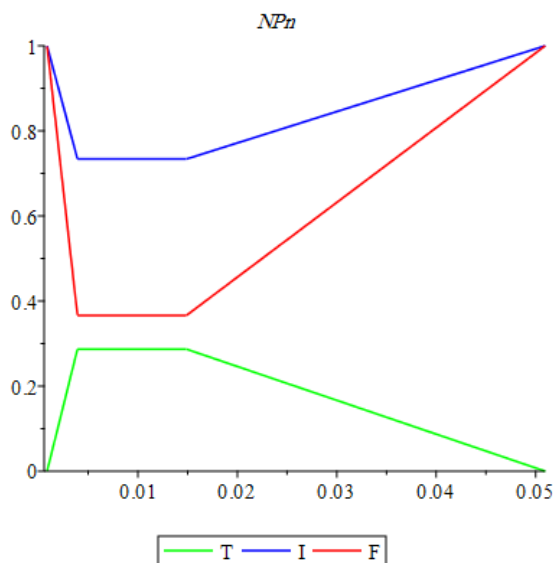


Fig14. NPn in single valued neutrosophic M/M/c/b queue

Neutrosophic performance measures using equations (3) to (6) will be:

$$NL_q = \left[\begin{matrix} [0.006424670732, 0.02349869451, 0.07434944242, 0.2184258058] , \\ 0.5090336259, 0.5139856353, 0.2064567181 \end{matrix} \right]$$

Which means that surely 50% the average number of customers in queue will range between 2.4% and 7.4% and will never be less than 0.64% or more than 21.8%.

We are also unsure with percentage of indeterminacy 51.4% of these results and we may be false by 20.6% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),lq);
```

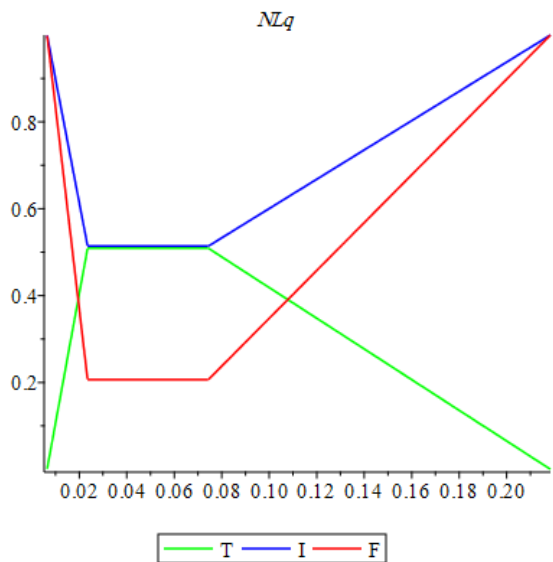


Fig15. NLq in single valued neutrosophic M/M/c/b queue

$$NL_s = \left[\begin{matrix} [0.3515279356, 0.5160637503, 0.7384051431, 1.092676261] , \\ 0.6358593744, 0.3920560678, 0.07821019297 \end{matrix} \right]$$

Which means that surely 63.6% the average number of customers in system will range between 52% and 74% and will never be less than 35% or more than 1.1.

We are also unsure with percentage of indeterminacy 39% of these results and we may be false by 7.8% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),ls);
```

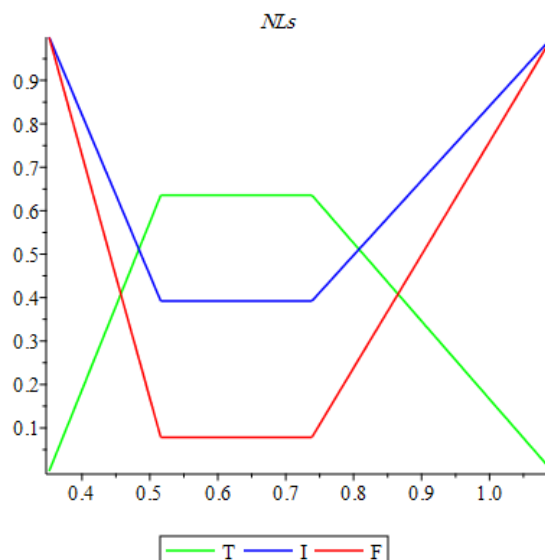


Fig16. NLs in single valued neutrosophic M/M/c/b queue

$$NW_q = \left[\begin{matrix} [0.001837193992, 0.007863695934, 0.03018867926, 0.1150779691] , \\ 0.1461031730, 0.8718954227, 0.5020886664 \end{matrix} \right]$$

Which means that surely 14.6% the mean waiting time in queue will range between 0.8% and 3% and will never be less than 0.2% or more than 11.5%.

We are also unsure with percentage of indeterminacy 87% of these results and we may be false by 50% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),wq);
```

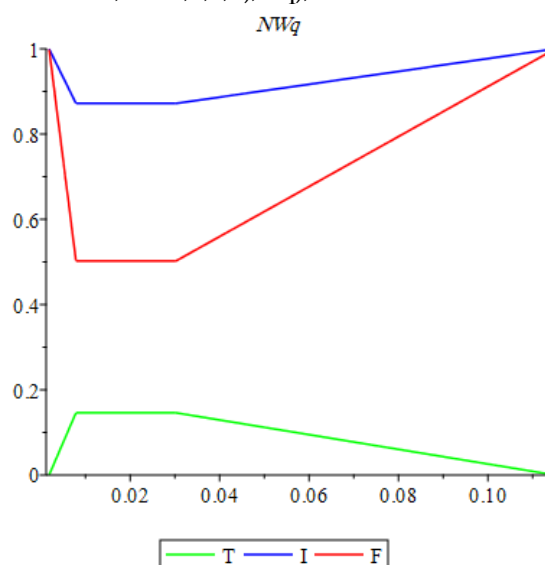


Fig17. NWq in single valued neutrosophic M/M/c/b queue

$$NW_s = \left[\begin{matrix} [0.1005226630, 0.1726976115, 0.2998203524, 0.5756781558] , \\ 0.1825047845, 0.8397569988, 0.4216199638 \end{matrix} \right]$$

Which means that surely 18% the mean waiting time in system will range between 29.9% and 83.9% and will never be less than 10% or more than 57.5%.

We are also unsure with percentage of indeterminacy 83.9% of these results and we may be false by 42% falsity degree.

```
SVTNQPlot(QueueingSystem(NLambda,NMu,4,2,4),ws);
```

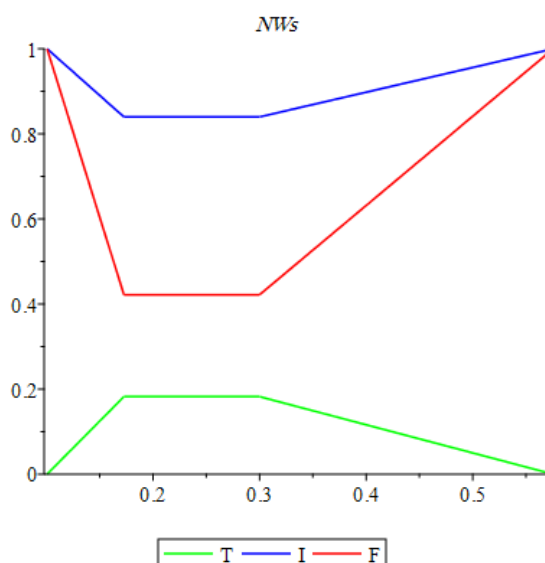


Fig18. NWs in single valued neutrosophic M/M/c/b queue

6. Conclusions

Ignoring indeterminacy may lead decision makers to make wrong decisions especially in controlling systems which is one of the most important applications of queueing theory. We found that neutrosophic queues are more reliable and applicable than crisp queues because of dealing with indeterminacy and uncertainty. We are looking forward to study and define more queueing systems in neutrosophic logic including tandem networks, open Jackson networks, balk queues, etc.

Funding: "This research received no external funding"

Acknowledgments: Authors are very grateful to the editor in chief and reviewers for their comments and suggestions, which is helpful in improving the paper.

Conflicts of Interest: "The authors declare no conflict of interest."

Appendix (Maple Code)

```
interface(warnlevel = 0):
with(plots):with(plottools):
SVTN := proc (a1, a2, a3, a4, wa, ua, ya)
return [[a1, a2, a3, a4], wa, ua, ya];
end proc:
SVTNNPlotT:=proc(n::list)
n1:=n[1][1];
n2:=n[1][2];
n3:=n[1][3];
n4:=n[1][4];
```



```

t:=n[2];

i:=n[3];

f:=n[4];

lt:=piecewise(n1<w and w<n2,t*(w-n1)/(n2-n1),undefined);

mt:=piecewise(n2<w and w<n3,t,undefined);

rt:=piecewise(n3<w and w<n4,t*(w-n4)/(n3-n4),undefined);

plot([lt(w),mt(w),rt(w)],w=n1..n4,color="green",legend=["T", "", ""],labels=["", ""]);

end proc;

SVTNNPlotI:=proc(n::list)

n1:=n[1][1];

n2:=n[1][2];

n3:=n[1][3];

n4:=n[1][4];

t:=n[2];

i:=n[3];

f:=n[4];

li:=piecewise(n1<u and u<n2,(n2-u+i*(u-n1))/(n2-n1),undefined);

mi:=piecewise(n2<u and u<n3,i,undefined);

ri:=piecewise(n3<u and u<n4,(u-n3+i*(n4-u))/(n4-n3),undefined);

plot([li(u),mi(u),ri(u)],u=n1..n4,color="blue",legend=["T", "", ""],labels=["", ""]);

end proc;

SVTNNPlotF:=proc(n::list)

n1:=n[1][1];

n2:=n[1][2];

n3:=n[1][3];

n4:=n[1][4];

t:=n[2];

i:=n[3];

```

```

f:=n[4];

lf:=piecewise(n1<y and y<n2,(n2-y+f*(y-n1))/(n2-n1),undefined);

mf:=piecewise(n2<y and y<n3,f,undefined);

rf:=piecewise(n3<y and y<n4,(y-n3+f*(n4-y))/(n4-n3),undefined);

plot([lf(y),mf(y),rf(y)],y=n1..n4,color="red",legend=["F","",""],labels=["","",""]);

end proc:

SVTNPlot:=proc(n::list,myTitle)

t:=SVTNNPlotT(n):

i:=SVTNNPlotI(n):

f:=SVTNNPlotF(n):

display([t, i, f],title=myTitle);

end proc:

SVTNQPlot:=proc(q::Matrix,property)

properties:=[p0,pn,lq,ls,wq,ws];

i:=ListTools[SearchAll](property,properties);

n:=rhs(op(convert(q[i],list)));

myTitle:=lhs(op(convert(q[i],list)));

SVTNPlot(n,myTitle);

end proc:

CrispNumberSS := proc (n) return SVTN(n, n, n, n, 0, 1, 1); end proc:

CrispNumberMD := proc (n) return SVTN(n, n, n, n, 1, 0, 0); end proc:

CrispNumber:=proc (n,NRho) return SVTN(n, n, n, n, NRho[2], NRho[3], NRho[4]); end proc:

SVTNSum := proc (t1, t2) x := t1; y := t2;

L1 := x[1][1]+y[1][1], x[1][2]+y[1][2], x[1][3]+y[1][3], x[1][4]+y[1][4];

U1 := x[2]+y[2]-x[2]*y[2], x[3]*y[3], x[4]*y[4]; [[L1], U1]; end proc:

SVTNSub := proc (t1, t2) x := t1; y := t2;

L1 := x[1][1]-y[1][4], x[1][2]-y[1][3], x[1][3]-y[1][2], x[1][4]-y[1][1];

U1 := x[2]+y[2]-x[2]*y[2], x[3]*y[3], x[4]*y[4]; [[L1], U1]; end proc:

```

```

SVTNMult := proc (t1, t2) x := t1; y := t2;

L1 := x[1][1]*y[1][1], x[1][2]*y[1][2], x[1][3]*y[1][3], x[1][4]*y[1][4];

U1 := x[2]*y[2], x[3]+y[3]-x[3]*y[3], x[4]+y[4]-x[4]*y[4]; [[L1], U1]; end proc;

SVTNScalarMult := proc (t1, PN) x := t1;

L1 := PN*x[1][1], PN*x[1][2], PN*x[1][3], PN*x[1][4];

U1 := 1-(1-x[2])^PN, x[3]^PN, x[4]^PN; [[L1], U1]; end proc;

SVTNDiv := proc (t1, t2) x := t1; y := t2;

L1 := x[1][1]/y[1][4], x[1][2]/y[1][3], x[1][3]/y[1][2], x[1][4]/y[1][1];

U1 := x[2]*y[2], x[3]+y[3]-x[3]*y[3], x[4]+y[4]-x[4]*y[4]; [[L1], U1]; end proc;

SVTNPower := proc (t1, PN) x := t1;

L1 := x[1][1]^PN, x[1][2]^PN, x[1][3]^PN, x[1][4]^PN;

U1 := x[2]^PN, 1-(1-x[3])^PN, 1-(1-x[4])^PN; [[L1], U1]; end proc;

SVTNSeries:=proc(x,n,NRho)

S:=CrispNumber(1,NRho);

for i from 1 by 1 to n-1 do

S:=SVTNSum(S,SVTNScalarMult(SVTNPower(x,i),1/(i!)));

end do;

S;

end proc;

SVTNSeriesmmcb:=proc(x,c,b)

Lq:=SVTNScalarMult(x,1/c);

for i from c+2 by 1 to b do

Lq:=SVTNSum(Lq,SVTNScalarMult(SVTNPower(SVTNScalarMult(x,1/c),i-c),i-c));

end do;

Lq;

end proc;

QueueingSystem:=overload(
[

```

```

proc(NLambda::list,NMu::list,n::integer,c::integer,b::integer) option overload;

NRho:=SVTNDiv(NLambda, NMu);

NP0:=evalf(SVTNDiv(CrispNumberMD(1),SVTNSum(SVTNSeries(NRho,c,NRho),SVTNMult(SVT
NScalarMult(SVTNPower(NRho,c),1/(c!)),SVTNDiv(SVTNSub(CrispNumberSS(1),SVTNPower(SV
TNScalarMult(NRho,1/c),b-c+1)),SVTNSub(CrispNumberSS(1),SVTNScalarMult(NRho,1/c))))));

if 0 <= n and n < c then
NPn:=evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,n),1/(n!)),NP0));

NPb:=evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,b),1/(b!)),NP0));

elif c <= n and n <= b then
NPn:=evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,n),1/(c!)),SVTNScalarMult(NP0,1/(c^(n
-c)))));

NPb:=evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,b),1/(c!)),SVTNScalarMult(NP0,1/(c^(b
-c)))));

end if;

NLq:=SVTNMult(SVTNMult(SVTNScalarMult(SVTNPower(NRho,c),1/c!),NP0),SVTNSeriesmmcb(
NRho,c,b));

NLambdae:= evalf(SVTNMult(NLambda,SVTNSub(CrispNumberSS(1),NPb)));

NLs:=evalf(SVTNSum(NLq,SVTNDiv(NLambdae,NMu)));

NWs:=evalf(SVTNDiv(NLs,NLambdae));

NWq:=evalf(SVTNDiv(NLq,NLambdae));

<'NP0'=NP0,
'NPn'=NPn,
'NLq'=NLq,
'NLs'=NLs,
'NWq'=NWq,
'NWs'=NWs>;

end proc,

#MMc

proc(NLambda,NMu,n,c) option overload;

NRho:=SVTNDiv(NLambda, NMu);

```

```
NP0:=evalf(SVTNDiv(CrispNumberMD(1),SVTNSum(SVTNSeries(NRho,c,NRho),SVTNDiv(SVTN
Power(NRho,c),SVTNScalarMult(SVTNSub(CrispNumberMD(1),SVTNScalarMult(NRho,1/c),c!))
));
```

```
NLq:=evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,c+1),1/(c!)),SVTNDiv(NP0,SVTNScalar
Mult(SVTNPower(SVTNSub(CrispNumberMD(1),SVTNScalarMult(NRho,1/c),2),c)))));
```

```
NLs:=evalf(SVTNSum(NLq,NRho));
```

```
NWs:=evalf(SVTNDiv(NLs,NLambda));
```

```
NWq:=evalf(SVTNDiv(NLq,NLambda));
```

```
if n>=0 and n < c then NPn:= evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,n),1/(n!)),NP0));
```

```
elif n>=c then NPn:=
```

```
evalf(SVTNMult(SVTNScalarMult(SVTNPower(NRho,n),1/(c!)),SVTNScalarMult(NP0,1/(c^(n-c))));
end if;
```

```
<'NP0'=NP0,
```

```
'NPn'=NPn,
```

```
'NLq'=NLq,
```

```
'NLs'=NLs,
```

```
'NWq'=NWq,
```

```
'NWs'=NWs>;
```

```
end proc,
```

```
#MM1
```

```
proc(NLambda,NMu,n) option overload;
```

```
NRho:=SVTNDiv(NLambda, NMu);
```

```
NLs:=evalf(SVTNDiv(NRho, SVTNSub(CrispNumberSS(1),NRho)));
```

```
NLq:=evalf(SVTNDiv(SVTNPower(NRho,2), SVTNSub(CrispNumberSS(1),NRho)));
```

```
NWs:=evalf(SVTNDiv(NLs, NLambda));
```

```
NWq:=evalf(SVTNDiv(NLq, NLambda));
```

```
NP0:=evalf(SVTNSub(CrispNumberSS(1),NRho));
```

```
NPn:=evalf(SVTNMult(NP0, SVTNPower(NRho, n)));
```

```
<'NP0'=NP0,
```

```
'NPn'=NPn,
```

'NLq'=NLq,

'NLs'=NLs,

'NWq'=NWq,

'NWs'=NWs>;

end proc

]

):

References

- [1] L. Zadeh, "Fuzzy Sets," *Information and Control*, vol. 8, no. 3, pp. 338-353, 1965.
- [2] A. Krassimir, "Intuitionistic Fuzzy Sets," *Fuzzy Sets and Systems*, vol. 20, no. 1, pp. 87-96, 1986.
- [3] F. Smarandache, "Indeterminacy in Neutrosophic Theories and their Applications," *International Journal of Neutrosophic Science*, vol. 15, no. 2, pp. 89-97, 2021.
- [4] H. Wang, F. Smarandache, Y. Zhang and R. Sunderraman, "Single Valued Neutrosophic Sets," *Multispace and Multistructure*, vol. 4, pp. 410-413, 2005.
- [5] E. AboElHamd, H. M. Shamma, M. Saleh and I. El-Khodary, "Neutrosophic Logic Theory and Applications," *Neutrosophic Sets and Systems*, vol. 41, pp. 30-51, 2021.
- [6] M. Abobala and A. Hatip, "An Algebraic Approach to Neutrosophic Euclidean Geometry," *Neutrosophic Sets and Systems*, vol. 43, pp. 114-123, 2021.
- [7] J. Ahmed, "LR-Type Fully Single-Valued Neutrosophic Linear Programming Problems," *Neutrosophic Sets and Systems*, vol. 46, pp. 416-444, 2021.
- [8] J. C. d. J. Arrias Añez, F. C. Bandera, L. R. Ayala Ayala, J. . C. N. Moncayo and M. D. L. C. S. Roig, "Neutrosophic Analysis of the Origin of Domestic Violence Sets," *Neutrosophic Sets and Systems*, vol. 44, pp. 26-34, 2021.
- [9] M. M. B. Benalcázar, B. D. N. Montenegro, M. J. Calderón Velásquez and J. R. C. Morillo, "Neutrosophic Statistic for Exploratory Analysis of the Data Provided by the Publications in the Social Sciences," *Neutrosophic Sets and Systems*, vol. 44, pp. 289-298, 2021.
- [10] A. R. Fernández, L. V. M. Rosales, O. G. A. Paspuel, W. B. J. López and A. R. S. León, "Neutrosophic Statistics for Project Management. Application to a Computer System Project," *Neutrosophic Sets and Systems*, vol. 44, pp. 308-314, 2021.
- [11] M. Jadid, A. A. Salama, R. Alhabib, H. E. Khalid and F. AL, "Neutrosophic Treatment of the Static Model of Inventory Management with Deficit," *International Journal of Neutrosophic Science*, vol. 18, no. 1, pp. 20-29, 2022.
- [12] D. V. P. Ruiz, R. A. Díaz Vásquez, B. E. V. Jadan and C. Y. D. Caballos, "Neutrosophic Statistics in the Strategic Planning of Information Systems," *Neutrosophic Sets and Systems*, vol. 44, pp. 402-410, 2021.

- [13] F. Smarandache, J. E. Ricardo, E. G. Caballero, M. Y. L. Vázquez and N. B. Hernández, "Delphi method for evaluating scientific research proposals in a neutrosophic environment," *Neutrosophic Sets and Systems*, vol. 34, 2020.
- [14] M. B. Zeina, O. Zeitouny, F. Masri, F. Kadoura and S. Broumi, "Operations on Single-Valued Trapezoidal Neutrosophic Numbers using (α, β, γ) -Cuts "Maple Package"," *International Journal of Neutrosophic Science*, vol. 15, no. 2, pp. 113-122, 2021.
- [15] M. B. Zeina and A. Hatip, " Neutrosophic Random Variables," *Neutrosophic Sets and Systems*, vol. 39, pp. 44-52, 2021.
- [16] C. Granados, " New Notions on Neutrosophic Random Variables," *Neutrosophic Sets and Systems*, vol. 47, pp. 286-297, 2021.
- [17] C. Granados and J. Sanabria, "On Independence Neutrosophic Random Variables," *Neutrosophic Sets and Systems*, vol. 47, pp. 541-557, 2021.
- [18] J. F. Shortle, J. M. Thompson, D. Gross and C. M. Harris, *Fundamentals of Queueing Theory*, Wiley Series in Probability and Statistics, 2018.
- [19] M. B. Zeina, K. Al-Kridi and M. T. Anan, "New Approach to FM/FM/1 Queue's Performance Measures," *King Abdulaziz University*, vol. 30, no. 1, 2017.
- [20] J. Kanyinda, R. Matendo and B. Lukata, "Computing Fuzzy Queueing Performance Measures by L-R Method," *ISPACS*, 2015.
- [21] S. Barak and M. S. Fallahnezhad, "Cost Analysis of Fuzzy Queueing Systems," *International Journal of Applied Operational Research*, 2005.
- [22] M. B. Zeina, "Erlang Service Queueing Model with Neutrosophic Parameters," *International Journal of Neutrosophic Science*, vol. 6, no. 2, pp. 106-112, 2020.
- [23] M. B. Zeina, "Neutrosophic Event-Based Queueing Model," *International Journal of Neutrosophic Science*, vol. 6, no. 1, pp. 48-55, 2020.
- [24] M. B. Zeina, "Neutrosophic M/M/1, M/M/c, M/M/1/b Queueing Systems," *Science Series*.
- [25] M. B. Zeina, "Linguistic Single Valued Neutrosophic M/M/1 Queue," *Science Series*.
- [26] I. Deli, "Operators on Single Valued Trapezoidal Neutrosophic Numbers and SVTN-Group Decision Making," *Neutrosophic Sets and Systems*, no. 22, 2018.

Received: Sep 7, 2022. Accepted: Dec 23, 2022