CAUCHY – Jurnal Matematika Murni dan Aplikasi
Volume 11 (2) (2026), Pages 2075–2095
p-ISSN: 2086-0382; e-ISSN: 2477-3344

Delay Tolerance Thresholds and Natural Recovery in a Fork–Join Production System over Max-Plus Algebra

Dewa Putu Wiadnyana Putra* and Marcellinus Andy Rudhito

Department of Mathematics Education, Universitas Sanata Dharma, Yogyakarta, Indonesia

Abstract

This study investigates server-specific delay tolerance and natural recovery in a three-server fork–join production system modeled with max-plus algebra. The system has two parallel servers synchronized by a join server and is analyzed in the canonical regime where the second-server self-loop is the unique critical circuit. Using max-plus spectral theory, critical-circuit margins, relative-delay recurrences, and computational validation, we examine a single additive event-time delay introduced after the nominal trajectory has entered the eigenvector regime. Every positive delay at the critical server propagates permanently, so its tolerance threshold is zero. At the two non-critical servers, exact finite thresholds are determined by the available margins between the critical circuit and competing non-critical paths. Delays not exceeding these thresholds are naturally absorbed without switching or control intervention, and a finite upper bound on the recovery time is established from a two-cycle contraction argument. Numerical tests across three admissible parameter sets confirm below-threshold recovery, boundary recovery, above-threshold propagation, and invariance with respect to the delay cycle and eigenvector translation. These results provide a precise event-level robustness characterization for the stated canonical fork–join regime.

Keywords: Critical circuit; Delay tolerance threshold; Fork–join system; Max-plus algebra; Natural recovery time.

1. Introduction

Modern production networks involve increasingly complex synchronized process flows. Components may be processed in parallel, subsequently merged, subjected to inspection processes that temporarily halt the flow, or assembled into batches at later stages [1, 2]. Such characteristics constitute a class of discrete-event systems. Max-plus algebra provides an effective theoretical framework for analyzing these dynamics through linear representations based on the operations of maximization and addition. In synchronized discrete-event systems, event times serve as state variables, while the operations of maximization and addition represent synchronization mechanisms and the accumulation of product processing times, respectively [3–6]. Furthermore, max-plus spectral theory provides a fundamental basis for analyzing asymptotic periodicity and critical graphs that characterize the paths dominating production dynamics [4, 7, 8].

The performance of servers in production systems often fails to operate consistently under nominal conditions. Variations in processing times, technical disruptions, and material delays

*Corresponding author. E-mail: dewa@usd.ac.id

Received: July 15, 2026 Revised: August 19, 2026 Accepted: September 8, 2026
DOI: https://doi.org/10.18860/cauchy.v11i2.45031

are among the factors that contribute to scheduling delays in certain servers. At the operational level, such perturbations may lead to delay propagation and increased operational costs [9, 10].

Earlier work by Putra and Rudhito [11] examined delay propagation in a three-server production system and used server switching, implemented by accelerating selected processing times, to restore schedules after propagative delays. The present article addresses a different question: the nominal max-plus dynamics are kept fixed, a single additive event-time delay is introduced at one specified server, and the objective is to determine the exact delay magnitude that can be absorbed naturally before the delay reaches and persists at the critical server. This distinction isolates intrinsic delay absorption under unchanged dynamics from recovery achieved through an external switching intervention.

Research on max-plus algebra in synchronization analysis has primarily focused on solution method and network optimization [12–14]. Previous studies have also examined robustness and perturbation bounds for max-plus systems. Global robustness of max-plus linear systems is formulated in terms of state variables remaining unchanged under bounded perturbations of system parameters [15, 16]. The q-step extension allows the relevant state components to become unchanged after a finite number of steps under such bounded parameter perturbations [17, 18]. These formulations therefore concern admissible perturbations of model parameters and state invariance properties. By contrast, the perturbation considered in this study leaves the matrix $\overline{A}$ unchanged and modifies one event time at one server by an additive amount δ. The quantities sought are consequently server-specific propagation thresholds and the number of cycles required for natural recovery. Switching max-plus frameworks and feedback or control studies address yet another objective: adapting modes, supervisory responses, or control decisions to disturbances and changing requirements [19–23].

Accordingly, the research gap addressed in this article is not a general absence of max-plus robustness theory, but the absence, within the specific canonical fork–join setting considered here, of closed-form server-level thresholds for a one-time additive event delay together with a finite-cycle recovery guarantee under unchanged nominal dynamics. The contribution is fourfold: (i) an additive single-delay model tied to a specified server and cycle; (ii) exact propagation thresholds $\Delta_2^* = 0$, $\Delta_3^* = s_{32}$, and $\Delta_1^* = s_{13} + s_{32}$ derived from critical-circuit margins; (iii) the natural-recovery bound $2\lceil \delta/\tau_{d_2} \rceil$ for threshold-admissible delays at non-critical servers; and (iv) computational validation of below-threshold, boundary, and above-threshold behavior across several admissible parameter sets. These claims are restricted to the canonical unique-critical-self-loop configuration and to delays introduced after the nominal trajectory has entered the max-plus eigenvector regime.

This article therefore aims to determine which servers in a three-server fork–join production system tolerate a one-time additive delay, to identify the exact threshold separating natural absorption from permanent propagation, and to bound the recovery time when the delay is non-propagative. The remainder of the paper is organized as follows. Section 2 presents the model and definitions, Section 3 develops the threshold and recovery analysis, and Section 4 concludes the paper. This organization separates the model specification from the threshold derivation, computational validation, and final synthesis.

2. Methods

This study is theoretical and analytical. The object of analysis is a three-server fork–join production system represented as a max-plus linear discrete-event system. The derivation proceeds by specifying the nominal max-plus model, defining a single delay perturbation, introducing delay tolerance and recovery concepts, and then analyzing delay trajectories under a canonical critical-circuit configuration. The proof strategy first identifies the canonical critical circuit, establishes persistence of the nominal eigenvector regime, derives a relative-delay recurrence, and then uses monotonicity and two-cycle contraction to obtain the threshold and recovery results. Computational checks subsequently evaluate the analytical predictions for below-threshold, boundary, and above-threshold delays and verify invariance with respect to the delay cycle and eigenvector translation. Throughout the threshold analysis, the perturbation is an additive event-time delay and the nominal trajectory is assumed to be in the eigenvector regime before the perturbation is applied.

Let $\mathbb{R}_{\max} = \mathbb{R} \cup \{\varepsilon\}$, where $\varepsilon = -\infty$. For $a, b \in \mathbb{R}_{\max}$, the max-plus addition and multiplication are defined by $a \oplus b = \max\{a, b\}$ and $a \otimes b = a + b$, respectively. Matrix-vector multiplication is defined using these operations in the usual way.

The three-server fork–join system consists of two parallel servers followed by one synchronization server, as illustrated in Fig. 1. The processing time of server i is denoted by $d_i \in \mathbb{R}_{>0}$ for $i = 1, 2, 3$, and the transfer or synchronization time is denoted by $t_j \in \mathbb{R}_{\geq 0}$ for $j = 1, 2, 3, 4, 5$. If $x_i(k)$ denotes the event time of server i in cycle k, then the cycle-to-cycle dynamics are first written componentwise as

$$x_1(k+1) = \max\{x_1(k) + d_1,\ x_3(k) + t_1 + d_3 + t_5\}, \tag{1}$$
$$x_2(k+1) = \max\{x_2(k) + d_2,\ x_3(k) + t_2 + d_3 + t_5\}, \tag{2}$$
$$x_3(k+1) = \max\{x_1(k) + d_1 + t_3,\ x_2(k) + d_2 + t_4,\ x_3(k) + d_3\}. \tag{3}$$

Eqs. (1)–(3) make explicit the dependencies represented in the state matrix. The first two equations combine the availability of each fork server with the inter-cycle release/feedback dependency from the preceding join-output cycle. The third equation represents the synchronization at server 3 of the two fork branches, together with the self-dependency of server 3. Thus Fig. 1 depicts the within-cycle process flow, whereas Eqs. (1)–(3) also record the cross-cycle dependencies needed for the discrete-event state recursion.

Collecting these three scalar recurrences gives the nominal max-plus system. Here $x(k) = [x_1(k), x_2(k), x_3(k)]^T$ denotes the event-time state vector at cycle k. The recursion is

$$x(k+1) = \overline{A} \otimes x(k), \tag{4}$$

where

$$\overline{A} = \begin{bmatrix} d_1 & \varepsilon & t_1 + d_3 + t_5 \\ \varepsilon & d_2 & t_2 + d_3 + t_5 \\ d_1 + t_3 & d_2 + t_4 & d_3 \end{bmatrix}. \tag{5}$$

Eqs. (4) and (5) define the nominal cycle-to-cycle recursion and its max-plus state matrix, respectively.

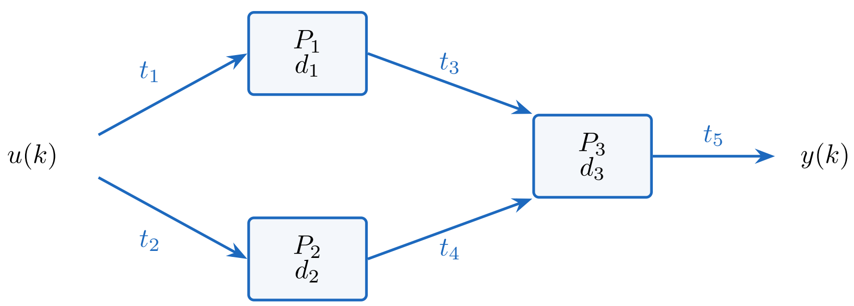

Fig. 1: Process diagram of the three-server fork–join production system. The arrows depict the within-cycle material/process flow; the cross-cycle release dependencies that generate the feedback entries of the max-plus state matrix are stated explicitly in Eqs. (1)–(3).

For notational simplicity, define

$$\alpha_1 = t_1 + d_3 + t_5, \quad \alpha_2 = t_2 + d_3 + t_5, \quad \beta_1 = d_1 + t_3, \quad \beta_2 = d_2 + t_4. \tag{6}$$

Then

$$\overline{A} = \begin{bmatrix} d_1 & \varepsilon & \alpha_1 \\ \varepsilon & d_2 & \alpha_2 \\ \beta_1 & \beta_2 & d_3 \end{bmatrix}. \tag{7}$$

The abbreviations introduced in Eq. (6) give the compact matrix representation in Eq. (7). The non-ε off-diagonal entries have a direct event-time interpretation. The entries $\overline{A}_{13} = \alpha_1 = t_1 + d_3 + t_5$ and $\overline{A}_{23} = \alpha_2 = t_2 + d_3 + t_5$ encode the cross-cycle release dependency from completion of the preceding join cycle to the next activation of fork servers 1 and 2, respectively. The entries $\overline{A}_{31} = \beta_1 = d_1 + t_3$ and $\overline{A}_{32} = \beta_2 = d_2 + t_4$ encode the two forward branch-to-join synchronization dependencies. Thus the "feedback" represented by $\overline{A}_{13}$ and $\overline{A}_{23}$ is a temporal release constraint between production cycles, not a reverse material-flow arrow in Fig. 1.

The precedence graph $G(\overline{A})$ associated with Eq. (7) is shown in Fig. 2. Its elementary circuits determine the possible max-plus eigenvalues and critical circuits of the system. This graph representation provides the structural basis for the circuit classification used in the threshold analysis.

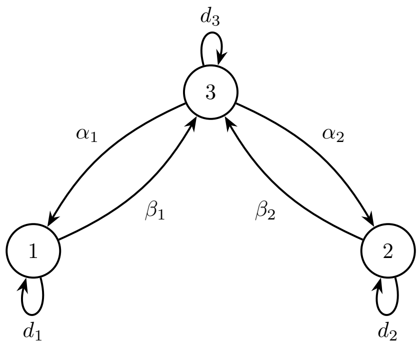

Fig. 2: Precedence graph of the matrix $\overline{A}$.

Definition 1 (Single additive delay)**.** Let $x(k_0)$ be the nominal event-time state at cycle k_0. A single additive delay of magnitude $\delta \in \mathbb{R}_{>0}$ at server i is defined componentwise by

$$\tilde{x}_j(k_0) = \begin{cases} x_i(k_0) + \delta, & j = i, \\ x_j(k_0), & j \neq i. \end{cases} \tag{8}$$

After the perturbation, the perturbed trajectory evolves under the same nominal dynamics,

$$\tilde{x}(k+1) = \overline{A} \otimes \tilde{x}(k), \qquad k \geq k_0. \tag{9}$$

Eqs. (8) and (9) specify, respectively, the one-time event perturbation and the unchanged post-perturbation dynamics. The delay is called non-propagative if there exists $N \geq k_0$ such that $\tilde{x}(k) = x(k)$ for every $k \geq N$.

Definition 2 (Delay tolerance threshold)**.** For each server $i \in \{1, 2, 3\}$, and within the nominal eigenvector regime specified below, define the set of positive tolerable delays by

$$\mathcal{T}_i = \left\{ \delta \in \mathbb{R}_{>0} : \begin{array}{l} \text{for every admissible delay cycle } k_0, \text{ the delay } \delta \text{ at server } i \\ \text{is non-propagative} \end{array} \right\}. \tag{10}$$

The delay tolerance threshold is then

$$\Delta_i^* = \sup(\{0\} \cup \mathcal{T}_i). \tag{11}$$

Thus Eq. (10) defines the admissible positive delays, while Eq. (11) converts that set into a server-specific threshold.

Definition 3 (Natural recovery time)**.** Suppose that a single additive delay of magnitude $\delta > 0$ occurs at server i at an admissible cycle k_0. If the perturbed trajectory eventually coincides permanently with the nominal trajectory, define

$$N_i^{\min}(\delta) = \min\left\{n \in \mathbb{Z}_{>0} : \widetilde{x}(k) = x(k) \text{ for every } k \geq k_0 + n\right\}. \tag{12}$$

If no such integer exists, set $N_i^{\min}(\delta) = \infty$. Thus Eq. (12) defines $N_i^{\min}(\delta)$ as the number of production cycles after the delay cycle k_0 required for permanent return to the nominal trajectory.

3. Results and Discussion

The analysis begins with the classification of the possible critical-circuit configurations of the system. This classification provides the structural basis for identifying the canonical critical case considered in the subsequent delay-tolerance and natural-recovery analysis.

3.1. Classification of critical circuits

The max-plus eigenvalue of an irreducible matrix is the maximum mean weight of all elementary circuits in the associated precedence graph. For the graph in Fig. 2, the elementary circuits are $c_1 = (1)$, $c_2 = (2)$, $c_3 = (3)$, $c_{13} = (1,3,1)$, and $c_{23} = (2,3,2)$. Writing $\mu(c)$ for the mean weight of a circuit c, their mean weights are d_1, d_2, d_3, $(\alpha_1 + \beta_1)/2$, and $(\alpha_2 + \beta_2)/2$, respectively.

Lemma 1. *The max-plus eigenvalue of the matrix $\overline{A}$ is*

$$\lambda(\overline{A}) = \max\left\{d_1, d_2, d_3, \frac{\alpha_1 + \beta_1}{2}, \frac{\alpha_2 + \beta_2}{2}\right\}. \tag{13}$$

Proof. Graph $G(\overline{A})$ in Fig. 2 is a strongly connected graph. It implies that the matrix $\overline{A}$ is irreducible. The max-plus eigenvalue $\lambda(\overline{A})$ is the maximum of the average weights of elementary circuits on $G(\overline{A})$. All possible elementary circuits on the graph $G(\overline{A})$ are

$$c_1 = (1), \qquad c_2 = (2), \qquad c_3 = (3), \qquad c_{13} = (1,3,1), \qquad c_{23} = (2,3,2).$$

The average weight of each circuit is

$$\mu(c_1) = d_1, \qquad \mu(c_2) = d_2, \qquad \mu(c_3) = d_3, \qquad \mu(c_{13}) = \frac{\alpha_1 + \beta_1}{2}, \qquad \mu(c_{23}) = \frac{\alpha_2 + \beta_2}{2}.$$

Hence,

$$\lambda(\overline{A}) = \max\left\{d_1, d_2, d_3, \frac{\alpha_1 + \beta_1}{2}, \frac{\alpha_2 + \beta_2}{2}\right\}.$$

□

Definition 4 (Critical server)**.** A server i is called critical if vertex i is included in at least one critical circuit of $G(\overline{A})$. The set of all critical servers is denoted by $\mathcal{S}_{\text{crit}}$.

From the definition, server 1 is critical if and only if $\lambda(\overline{A}) = d_1$ or $\lambda(\overline{A}) = (\alpha_1 + \beta_1)/2$. Server 2 is critical if and only if $\lambda(\overline{A}) = d_2$ or $\lambda(\overline{A}) = (\alpha_2 + \beta_2)/2$. Server 3 is critical if and only if $\lambda(\overline{A}) = d_3$, $\lambda(\overline{A}) = (\alpha_1 + \beta_1)/2$, or $\lambda(\overline{A}) = (\alpha_2 + \beta_2)/2$.

Equation (13) gives several possible critical-circuit configurations. The critical circuit may correspond to a single elementary circuit of $G(\overline{A})$, although multiple critical circuits may also exist. There are $2^5 - 1$ possible nonempty subsets of elementary circuits according to the power set. The conditions in the definition of a critical server reduce the feasible space, because some sets of critical circuits cannot occur. For example, the family

$$\mathcal{A} = \{\{c_1, c_3\}, \{c_2, c_3\}, \{c_1, c_2, c_3\}, \{c_1, c_3, c_{23}\}, \{c_2, c_3, c_{13}\}, \{c_1, c_2, c_3, c_{13}\}, \{c_1, c_2, c_3, c_{23}\}\}$$

is infeasible as a family of critical circuits of $G(\overline{A})$. Therefore, the total number of possible sets of critical circuits is 24. Under the symmetry of the fork branches, the feasible critical-circuit configurations can be grouped into 15 classes, as shown in Table 1.

Table 1: Class of critical circuit configurations

Class	Set of critical circuits
K_1	$\{\{c_1\}, \{c_2\}\}$
K_2	$\{\{c_3\}\}$
K_3	$\{\{c_{13}\}, \{c_{23}\}\}$
K_4	$\{c_1, c_2\}$
K_5	$\{\{c_1, c_{13}\}, \{c_2, c_{23}\}\}$
K_6	$\{\{c_1, c_{23}\}, \{c_2, c_{13}\}\}$
K_7	$\{\{c_3, c_{13}\}, \{c_3, c_{23}\}\}$
K_8	$\{c_{13}, c_{23}\}$
K_9	$\{\{c_1, c_2, c_{13}\}, \{c_1, c_2, c_{23}\}\}$
K_{10}	$\{\{c_1, c_3, c_{13}\}, \{c_2, c_3, c_{23}\}\}$
K_{11}	$\{\{c_1, c_{13}, c_{23}\}, \{c_2, c_{13}, c_{23}\}\}$
K_{12}	$\{c_3, c_{13}, c_{23}\}$
K_{13}	$\{c_1, c_2, c_{13}, c_{23}\}$
K_{14}	$\{\{c_1, c_3, c_{13}, c_{23}\}, \{c_2, c_3, c_{13}, c_{23}\}\}$
K_{15}	$\{c_1, c_2, c_3, c_{13}, c_{23}\}$

The 15 classes in Table 1 provide structural context for the possible critical-circuit configurations; the present article does not claim delay-threshold results for all of them. The analysis is restricted to class K_1, in which the unique critical circuit is a self-loop on one of the two parallel fork servers, namely $\{c_1\}$ or $\{c_2\}$. This restriction keeps the subsequent threshold statements tied explicitly to the canonical unique-critical-self-loop regime.

Class K_1 is selected as the focus because it is the canonical configuration in which criticality is localized at a single fork-server self-loop while all competing circuits are strictly non-critical. This structure isolates the transmission of a localized event-time delay relative to a single critical server and yields strictly positive critical-circuit margins that can be used to derive closed-form delay recurrences, exact server-specific tolerance thresholds, and a finite natural-recovery bound. It therefore provides a mathematically tractable baseline for identifying the threshold mechanism before considering configurations with critical two-vertex circuits or multiple critical circuits, whose delay interactions require separate analysis.

Owing to the symmetry of the fork section, it is sufficient to analyze $\{c_2\}$ as the canonical representative. The corresponding result for the branch-interchanged case $\{c_1\}$ follows by the symmetry transformation established below. Other single- or multiple-critical-circuit classes are outside the scope of the current paper.

3.2. Branch symmetry and canonical critical case

Theorem 1 (Branch symmetry). *Let*

$$J = \{x_1, x_2, d_1, d_2, t_1, t_2, t_3, t_4, \alpha_1, \alpha_2, \beta_1, \beta_2\}.$$

Define the involution S on J by the branch-index permutation $1 \leftrightarrow 2$ and $3 \leftrightarrow 4$. Then the *matrix $\overline{A}$ is invariant up to permutation under the transformation S.*

Proof. The image of the matrix $\overline{A}$ in Eq. (5) under the transformation S is

$$S(\overline{A}) = \begin{bmatrix} d_2 & \varepsilon & t_2+d_3+t_5 \\ \varepsilon & d_1 & t_1+d_3+t_5 \\ d_2+t_4 & d_1+t_3 & d_3 \end{bmatrix} = \begin{bmatrix} d_2 & \varepsilon & \alpha_2 \\ \varepsilon & d_1 & \alpha_1 \\ \beta_2 & \beta_1 & d_3 \end{bmatrix}.$$

If

$$P = \begin{bmatrix} \varepsilon & 0 & \varepsilon \\ 0 & \varepsilon & \varepsilon \\ \varepsilon & \varepsilon & 0 \end{bmatrix},$$

then

$$\overline{A} = P \otimes S(\overline{A}) \otimes P.$$

Because P is the max-plus permutation matrix that exchanges the first two coordinates and satisfies $P^{-1} = P$, the identity above shows that $\overline{A}$ and $S(\overline{A})$ are permutation-similar. Hence exchanging the two fork branches preserves the max-plus spectral and delay-propagation structure used below. □

We now assume that the self-loop c_2 is the unique critical circuit. Thus,

$$d_2 > \max\left\{d_1, d_3, \frac{\alpha_1+\beta_1}{2}, \frac{\alpha_2+\beta_2}{2}\right\}. \tag{14}$$

Condition (14) is the canonical K_1 case treated in the remainder of the analysis. All threshold and recovery statements below are to be understood under this unique-critical-self-loop condition, unless explicitly stated otherwise. The relevant positive margins are defined by

$$\sigma_1 = d_2 - d_1, \quad \sigma_3 = d_2 - d_3, \quad s_{13} = 2d_2 - \alpha_1 - \beta_1, \quad s_{32} = d_2 - \alpha_2 - t_4. \tag{15}$$

These margins measure the separation between the critical self-loop and the other elementary paths that can transmit a delay.

Lemma 2. *If d_2 is the max-plus eigenvalue of $\overline{A}$ under Eq. (14), then*

$$v = \begin{bmatrix} \alpha_1+t_4-d_2+s \\ s \\ t_4+s \end{bmatrix}, \qquad s \in \mathbb{R}, \tag{16}$$

is a max-plus eigenvector associated with $\lambda = d_2$.

Proof. Suppose $v = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}^T$ is a max-plus eigenvector corresponding to the max-plus eigenvalue $\lambda = d_2$. Then

$$\begin{bmatrix} d_1 & \varepsilon & \alpha_1 \\ \varepsilon & d_2 & \alpha_2 \\ \beta_1 & \beta_2 & d_3 \end{bmatrix} \otimes \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = d_2 \otimes \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix},$$

or equivalently

$$\begin{bmatrix} \max\{d_1+v_1, \alpha_1+v_3\} \\ \max\{d_2+v_2, \alpha_2+v_3\} \\ \max\{\beta_1+v_1, \beta_2+v_2, d_3+v_3\} \end{bmatrix} = \begin{bmatrix} d_2+v_1 \\ d_2+v_2 \\ d_2+v_3 \end{bmatrix}.$$

The first component of the vector implies

$$\max\{d_1 + v_1, \alpha_1 + v_3\} = d_2 + v_1.$$

Then

$$\max\{d_1 + v_1, \alpha_1 + v_3\} = \alpha_1 + v_3$$

because $d_2 > d_1$. Therefore,

$$\alpha_1 + v_3 = d_2 + v_1. \tag{17}$$

The second component does not provide much information because

$$\max\{d_2 + v_2, \alpha_2 + v_3\} = d_2 + v_2.$$

The last component yields the following relationship:

$$\max\{\beta_1 + v_1, \beta_2 + v_2, d_3 + v_3\} = d_2 + v_3.$$

The equality

$$\max\{\beta_1 + v_1, \beta_2 + v_2, d_3 + v_3\} = d_3 + v_3$$

is impossible because $d_2 > d_3$. Hence,

$$\max\{\beta_1 + v_1, \beta_2 + v_2\} = d_2 + v_3.$$

Assume that $\beta_1 + v_1 = d_2 + v_3$. Substituting this equality into Eq. (17) gives

$$\beta_1 + (\alpha_1 + v_3 - d_2) = d_2 + v_3,$$

so

$$\alpha_1 + \beta_1 = 2d_2.$$

This is impossible because $s_{13} > 0$. Therefore,

$$\max\{\beta_1 + v_1, \beta_2 + v_2, d_3 + v_3\} = \beta_2 + v_2 = d_2 + v_3.$$

Since $\beta_2 = d_2 + t_4$, we obtain

$$t_4 + v_2 = v_3. \tag{18}$$

Choose $v_2 = s \in \mathbb{R}$. From Eqs. (17) and (18), we obtain

$$v_1 = \alpha_1 + t_4 + s - d_2 \qquad \text{and} \qquad v_3 = t_4 + s.$$

Hence, the max-plus eigenvector corresponding to the max-plus eigenvalue $\lambda = d_2$ is precisely the vector stated in Eq. (16). □

Assumption 1 (Nominal eigenvector regime)**.** There exists a cycle k_s and a scalar $s_s \in \mathbb{R}$ such that the nominal state has reached an eigenvector associated with the unique eigenvalue d_2,

$$x(k_s) = \begin{bmatrix} \alpha_1 + t_4 - d_2 + s_s \\ s_s \\ t_4 + s_s \end{bmatrix}. \tag{19}$$

The nominal state in Eq. (19) fixes the eigenvector regime used throughout the perturbation analysis. The single additive delay considered in this article is introduced only at a cycle $k_0 \geq k_s$. No claim is made for a delay introduced during an arbitrary transient state before the nominal trajectory reaches this regime.

Lemma 3 (Persistence of the nominal eigenvector regime)**.** *Under Eq.* (14) *and Assumption 1, for every integer* $m \geq 0$*,*

$$x(k_s + m) = d_2^{\otimes m} \otimes x(k_s) = \begin{bmatrix} \alpha_1 + t_4 - d_2 + s_s + md_2 \\ s_s + md_2 \\ t_4 + s_s + md_2 \end{bmatrix}. \tag{20}$$

Hence the nominal trajectory remains in the same max-plus eigenvector class for every $k \geq k_s$*.*

Proof. By the preceding eigenvector lemma,

$$\overline{A} \otimes x(k_s) = d_2 \otimes x(k_s).$$

For $m = 0$, Eq. (20) is immediate. Assume it holds for some $m \geq 0$. Using max-plus homogeneity,

$$\begin{aligned} x(k_s + m + 1) &= \overline{A} \otimes x(k_s + m) \\ &= \overline{A} \otimes (d_2^{\otimes m} \otimes x(k_s)) \\ &= d_2^{\otimes m} \otimes (\overline{A} \otimes x(k_s)) \\ &= d_2^{\otimes (m+1)} \otimes x(k_s). \end{aligned}$$

Thus Eq. (20) follows by induction. In ordinary arithmetic, multiplication by $d_2^{\otimes m}$ adds md_2 to every finite component, which gives the displayed vector form. □

3.3. Delay trajectory in the canonical case

Let $\widetilde{x}(k)$ be the perturbed trajectory generated from an additive delay at an admissible cycle $k_0 \geq k_s$. Define the componentwise delay magnitudes by

$$p(k) = \widetilde{x}_1(k) - x_1(k), \quad q(k) = \widetilde{x}_2(k) - x_2(k), \quad r(k) = \widetilde{x}_3(k) - x_3(k). \tag{21}$$

Because $\widetilde{x}(k_0) \geq x(k_0)$ componentwise and the max-plus map $z \mapsto \overline{A} \otimes z$ is monotone, it follows that

$$p(k) \geq 0, \qquad q(k) \geq 0, \qquad r(k) \geq 0, \qquad k \geq k_0. \tag{22}$$

The non-negativity property in Eq. (22) is used throughout the delay-propagation analysis below. Immediately after a single delay of magnitude δ at cycle k_0, the initial delay vector is

$$(p(k_0), q(k_0), r(k_0)) = \begin{cases} (\delta, 0, 0), & \text{delay at server 1,} \\ (0, \delta, 0), & \text{delay at server 2,} \\ (0, 0, \delta), & \text{delay at server 3.} \end{cases} \tag{23}$$

Eqs. (21)–(23) define the relative-delay state, its non-negativity, and the three possible single-server initial perturbations. The next lemma is therefore stated under the unique-critical-self-loop condition (14), Assumption 1, and the nominal-regime persistence established in Lemma 3.

Lemma 4 (Delay recurrence)**.** *Assume Eq.* (14) *and Assumption 1. By Lemma 3, the nominal trajectory satisfies the eigenvector representation in Eq.* (20) *and the relations in Eqs.* (17) *and* (18)*. If a single additive delay is introduced at any admissible cycle* $k_0 \geq k_s$*,* *then for every $k \geq k_0$ the delay magnitudes satisfy*

$$p(k+1) = \max\{p(k) - \sigma_1, r(k)\}, \tag{24}$$
$$q(k+1) = \max\{q(k), r(k) - s_{32}\}, \tag{25}$$
$$r(k+1) = \max\{p(k) - s_{13}, q(k), r(k) - \sigma_3\}. \tag{26}$$

Proof. For the first component, Eqs. (1) and (9) give

$$\widetilde{x}_1(k+1) = \max\{\widetilde{x}_1(k) + d_1, \widetilde{x}_3(k) + \alpha_1\}.$$

By Eqs. (17) and (18), the nominal eigenvector regime satisfies

$$x_1(k+1) = x_3(k) + \alpha_1 = x_1(k) + d_2.$$

Hence

$$\begin{aligned} p(k+1) &= \widetilde{x}_1(k+1) - x_1(k+1) \\ &= \max\{x_1(k) + p(k) + d_1,\ x_3(k) + r(k) + \alpha_1\} - (x_3(k) + \alpha_1) \\ &= \max\{p(k) + d_1 - d_2,\ r(k)\} \\ &= \max\{p(k) - \sigma_1, r(k)\}. \end{aligned}$$

For the second component,

$$\widetilde{x}_2(k+1) = \max\{\widetilde{x}_2(k) + d_2, \widetilde{x}_3(k) + \alpha_2\},$$

whereas the nominal eigenvector relations give $x_2(k+1) = x_2(k) + d_2$ and $x_3(k) = x_2(k) + t_4$. Therefore

$$\begin{aligned} q(k+1) &= \max\{x_2(k) + q(k) + d_2,\ x_3(k) + r(k) + \alpha_2\} - (x_2(k) + d_2) \\ &= \max\{q(k),\ r(k) + \alpha_2 + t_4 - d_2\} \\ &= \max\{q(k), r(k) - s_{32}\}. \end{aligned}$$

For the third component,

$$\widetilde{x}_3(k+1) = \max\{\widetilde{x}_1(k) + \beta_1, \widetilde{x}_2(k) + \beta_2, \widetilde{x}_3(k) + d_3\}.$$

Since $\beta_2 = d_2 + t_4$, the steady relations imply

$$x_3(k+1) = x_2(k) + \beta_2.$$

Moreover, using the nominal eigenvector relations and the definitions in Eq. (15),

$$x_1(k) + \beta_1 = x_2(k) + \beta_2 - s_{13}, \tag{27}$$
$$x_3(k) + d_3 = x_2(k) + \beta_2 - \sigma_3. \tag{28}$$

Using Eqs. (27)–(28), consequently,

$$\begin{aligned} r(k+1) &= \widetilde{x}_3(k+1) - x_3(k+1) \\ &= \max\{p(k) - s_{13}, q(k), r(k) - \sigma_3\}. \end{aligned}$$

Thus Eqs. (24)–(26) hold for every $k \geq k_0$. □

Lemma 5 (Two-cycle contraction and finite recovery)**.** *Let*

$$M(k) = \max\{p(k), q(k), r(k)\} \quad \text{and} \quad \tau_{d_2} = \min\{\sigma_1, \sigma_3, s_{13}\} > 0. \tag{29}$$

If $q(k) = 0$ for every $k \geq k_0$, then $M(k)$ is non-increasing and satisfies

$$M(k+2) \leq \max\{M(k) - \tau_{d_2}, 0\}, \qquad k \geq k_0. \tag{30}$$

Consequently, if $M(k_0) = M_0$, then

$$M(k_0 + 2m) \leq \max\{M_0 - m\tau_{d_2}, 0\}, \qquad m \in \mathbb{Z}_{\geq 0}, \tag{31}$$

and the perturbed trajectory reaches the nominal trajectory in finitely many cycles.

Proof. When $q(k) = 0$, Lemma 4 gives

$$p(k+1) = \max\{p(k) - \sigma_1, r(k)\} \leq M(k),$$

and

$$r(k+1) = \max\{p(k) - s_{13}, 0, r(k) - \sigma_3\} \leq M(k).$$

Hence

$$M(k+1) = \max\{p(k+1), r(k+1)\} \leq M(k),$$

so $M(k)$ is non-increasing. Moreover, by the definition of τ_{d_2} in Eq. (29),

$$r(k+1) \leq \max\{M(k) - s_{13}, 0, M(k) - \sigma_3\} \leq \max\{M(k) - \tau_{d_2}, 0\}.$$

Using $p(k+1) \leq M(k)$ and the recurrence once more,

$$\begin{aligned} p(k+2) &= \max\{p(k+1) - \sigma_1, r(k+1)\} \\ &\leq \max\{M(k) - \sigma_1, M(k) - \tau_{d_2}, 0\} \\ &\leq \max\{M(k) - \tau_{d_2}, 0\}, \end{aligned}$$

while

$$\begin{aligned} r(k+2) &= \max\{p(k+1) - s_{13}, 0, r(k+1) - \sigma_3\} \\ &\leq \max\{M(k) - s_{13}, 0, M(k) - \tau_{d_2} - \sigma_3\} \\ &\leq \max\{M(k) - \tau_{d_2}, 0\}. \end{aligned}$$

Because $q(k+2) = 0$, Eq. (30) follows. Iterating it yields Eq. (31). Taking

$$m = \left\lceil \frac{M_0}{\tau_{d_2}} \right\rceil$$

gives $M(k_0 + 2m) = 0$. Since $M(k)$ is non-increasing and nonnegative, $M(k) = 0$ for every later cycle as well. Thus recovery is finite. $\square$

3.4. Delay tolerance thresholds and natural recovery

Lemma 6 (Persistence of delay at the critical server). *Under Eq. (14) and Assumption 1, the delay at server 2 satisfies*

$$q(k+1) \geq q(k), \qquad k \geq k_0. \tag{32}$$

Consequently, if $q(k_1) > 0$ at any cycle $k_1 \geq k_0$, then the perturbed trajectory can never return to the nominal trajectory.

Proof. From the delay recurrence (25),

$$q(k+1) = \max\{q(k), r(k) - s_{32}\} \geq q(k).$$

This establishes Eq. (32), so $q(k)$ is non-decreasing. If $q(k_1) > 0$, then $q(k) \geq q(k_1) > 0$ for every $k \geq k_1$. Hence $\widetilde{x}_2(k) \neq x_2(k)$ for all such k, so the perturbation is propagative by the definition of a non-propagative delay. □

Theorem 2 (Zero tolerance at the critical server). *Every positive additive delay at server 2 is propagative. Consequently,*

$$\Delta_2^* = 0. \tag{33}$$

Proof. Let a delay $\delta > 0$ be introduced at server 2 at an admissible cycle k_0. By Eq. (23),

$$p(k_0) = 0, \qquad q(k_0) = \delta, \qquad r(k_0) = 0.$$

Since $q(k_0) = \delta > 0$, Lemma 6 implies that $q(k) \geq \delta > 0$ for every $k \geq k_0$. Therefore the perturbed trajectory never coincides permanently with the nominal trajectory, and every positive delay at server 2 is propagative. By the threshold definition (11), whose admissible set always contains 0, it follows that $\Delta_2^* = 0$, which establishes Eq. (33). □

Theorem 3 (Exact thresholds at the non-critical servers). *Under the canonical condition (14) and Assumption 1, the exact delay tolerance thresholds at servers 3 and 1 are*

$$\Delta_3^* = s_{32}, \qquad \Delta_1^* = s_{13} + s_{32}. \tag{34}$$

Proof. We prove the two statements separately.

Server 3. Suppose a delay $\delta > 0$ is introduced at server 3 at cycle k_0. Then

$$p(k_0) = q(k_0) = 0, \qquad r(k_0) = \delta.$$

The recurrence (25) gives

$$q(k_0 + 1) = \max\{0, \delta - s_{32}\}. \tag{35}$$

By Eq. (35), if $\delta > s_{32}$, then $q(k_0 + 1) = \delta - s_{32} > 0$. Lemma 6 therefore shows that the delay is propagative.

Now assume $0 < \delta \leq s_{32}$. We prove by induction the invariant

$$q(k) = 0, \qquad 0 \leq p(k) \leq \delta, \qquad 0 \leq r(k) \leq \delta, \qquad k \geq k_0. \tag{36}$$

The invariant in Eq. (36) holds at $k = k_0$. Suppose it holds at some $k \geq k_0$. Since $r(k) \leq \delta \leq s_{32}$, Eq. (25) gives

$$q(k+1) = \max\{0, r(k) - s_{32}\} = 0.$$

Furthermore, because $\sigma_1, \sigma_3, s_{13} > 0$,

$$p(k+1) = \max\{p(k) - \sigma_1, r(k)\} \leq \delta$$

and

$$r(k+1) = \max\{p(k) - s_{13}, 0, r(k) - \sigma_3\} \leq \delta.$$

Thus the invariant holds for all $k \geq k_0$. In particular, the critical component remains undelayed. Lemma 5 then implies finite recovery to the nominal trajectory. Hence every $\delta \leq s_{32}$ is non-propagative, whereas every $\delta > s_{32}$ is propagative, and therefore

$$\Delta_3^* = s_{32}.$$

Server 1. Suppose a delay $\delta > 0$ is introduced at server 1 at cycle k_0, so

$$p(k_0) = \delta, \qquad q(k_0) = r(k_0) = 0.$$

At the first subsequent cycle,

$$r(k_0+1) = \max\{\delta - s_{13}, 0\}, \qquad q(k_0+1) = 0. \tag{37}$$

At the next cycle,

$$q(k_0+2) = \max\{0, \delta - s_{13} - s_{32}\}. \tag{38}$$

Eqs. (37)–(38) show that, if $\delta > s_{13} + s_{32}$, then $q(k_0+2) > 0$, and Lemma 6 implies propagation.

It remains to prove recovery when $0 < \delta \leq s_{13} + s_{32}$. Define

$$R_\delta = \max\{\delta - s_{13}, 0\}. \tag{39}$$

For R_δ defined in Eq. (39), the threshold assumption gives $R_\delta \leq s_{32}$. We show by induction that, for every $k \geq k_0$,

$$q(k) = 0, \qquad 0 \leq p(k) \leq \delta, \qquad 0 \leq r(k) \leq R_\delta. \tag{40}$$

The invariant in Eq. (40) holds at $k = k_0$ because $q(k_0) = r(k_0) = 0$. Assume it holds at cycle k. Since $r(k) \leq R_\delta \leq s_{32}$,

$$q(k+1) = \max\{0, r(k) - s_{32}\} = 0.$$

Also,

$$p(k+1) = \max\{p(k) - \sigma_1, r(k)\} \leq \max\{\delta, R_\delta\} = \delta,$$

and

$$\begin{aligned} r(k+1) &= \max\{p(k) - s_{13}, 0, r(k) - \sigma_3\} \\ &\leq \max\{\delta - s_{13}, 0, R_\delta\} = R_\delta. \end{aligned}$$

Thus the invariant is preserved. Hence $q(k) = 0$ for every $k \geq k_0$, and Lemma 5 guarantees finite recovery. Consequently every $\delta \leq s_{13} + s_{32}$ is non-propagative, while every larger delay is propagative. Therefore

$$\Delta_1^* = s_{13} + s_{32}.$$

This proves Eq. (34). □

Theorem 4 (Natural recovery bound). *Under the canonical condition* (14) *and Assumption 1, a positive delay at the critical server satisfies*

$$N_2^{\min}(\delta) = \infty, \qquad \delta > 0. \tag{41}$$

For a single additive delay at a non-critical server $i \in \{1, 3\}$ *satisfying* $0 < \delta \le \Delta_i^*$,

$$N_i^{\min}(\delta) \le 2 \left\lceil \frac{\delta}{\tau_{d_2}} \right\rceil, \qquad \tau_{d_2} = \min\{\sigma_1, \sigma_3, s_{13}\}. \tag{42}$$

Proof. For server 2, Theorem 2 shows that every $\delta > 0$ is propagative. By Definition 3, the trajectory never recovers permanently, so $N_2^{\min}(\delta) = \infty$, which proves Eq. (41).
Now let $i \in \{1, 3\}$ and $0 < \delta \le \Delta_i^*$. Theorem 3 and its proof show that $q(k) = 0$ for every $k \ge k_0$. At the delay cycle, exactly one component has delay δ, hence

$$M(k_0) = \max\{p(k_0), q(k_0), r(k_0)\} = \delta.$$

Applying Eq. (31) with $M_0 = \delta$ gives

$$M(k_0 + 2m) \le \max\{\delta - m\tau_{d_2}, 0\}.$$

Choose

$$m = \left\lceil \frac{\delta}{\tau_{d_2}} \right\rceil.$$

Then $m\tau_{d_2} \ge \delta$, so $M(k_0 + 2m) = 0$. Because M is non-increasing and the zero delay state is invariant under the recurrence, the perturbed and nominal trajectories coincide for all subsequent cycles. Therefore

$$N_i^{\min}(\delta) \le 2m = 2 \left\lceil \frac{\delta}{\tau_{d_2}} \right\rceil.$$

This establishes Eq. (42). The factor 2 lies outside the ceiling because Lemma 5 guarantees a reduction of at least τ_{d_2} only after each *pair* of cycles. □

Table 2 summarizes the analytical consequences. It shows that the critical server has no delay tolerance, while the non-critical servers may recover naturally as long as the delay remains below the corresponding threshold. The same table records the finite recovery-time upper bound established for threshold-admissible delays at the two non-critical servers.

Table 2: Summary of delay tolerance and natural recovery in the canonical case

Server	Critical status	Threshold	Natural recovery time
1	Non-critical	$s_{13} + s_{32}$	$N_1^{\min}(\delta) \le 2 \left\lceil \frac{\delta}{\tau_{d_2}} \right\rceil$
2	Critical	0	$N_2^{\min}(\delta) = \infty$
3	Non-critical	s_{32}	$N_3^{\min}(\delta) \le 2 \left\lceil \frac{\delta}{\tau_{d_2}} \right\rceil$

3.5. Computational validation

Consider the parameters

$$d_1 = 5, \quad d_2 = 8, \quad d_3 = 3, \quad t_1 = 1, \quad t_2 = 0, \quad t_3 = 2, \quad t_4 = 1, \quad t_5 = 1.$$

Then $\alpha_1 = 5$, $\alpha_2 = 4$, $\beta_1 = 7$, and $\beta_2 = 9$. The average weights of the elementary circuits are

$$\mu(c_1) = 5, \quad \mu(c_2) = 8, \quad \mu(c_3) = 3, \quad \mu(c_{13}) = 6, \quad \mu(c_{23}) = 6.5.$$

Thus c_2 is the unique critical circuit. Moreover, $\sigma_1 = 3$, $\sigma_3 = 5$, $s_{13} = 4$, $s_{32} = 3$, and $\tau_{d_2} = 3$. Therefore, $\Delta_2^* = 0$, $\Delta_3^* = 3$, and $\Delta_1^* = 7$.

Representative delay trajectories for the critical server and the two non-critical servers are reported in Tables 3, 4, and 5. These trajectories illustrate the distinct behaviors predicted by the analytical results: persistence at the critical server and finite recovery at threshold-admissible non-critical servers. They also provide concrete cycle-by-cycle examples before the broader boundary validation across multiple parameter sets.

Table 3: Delay trajectory for $\delta = 1$ at server 2

Cycle	$(p(k), q(k), r(k))$	Description
k_0	$(0, 1, 0)$	Delay starts at server 2
$k_0 + 1$	$(0, 1, 1)$	Delay propagates to server 3
$k_0 + 2$	$(1, 1, 1)$	Propagation reaches all servers
$k_0 + 3$	$(1, 1, 1)$	All servers remain delayed by δ
$\vdots$	$(1, 1, 1)$	Delay propagation persists

Table 4: Delay trajectory for $\delta = 3$ at server 3

Cycle	$(p(k), q(k), r(k))$	Description
k_0	$(0, 0, 3)$	Delay starts at server 3
$k_0 + 1$	$(3, 0, 0)$	Delay shifts from server 3 to server 1
$k_0 + 2$	$(0, 0, 0)$	System returns to nominal operation

For server 3 with $\delta = 3$, the system recovers within two cycles,

$$N_3^{\min}(3) = 2 \leq 2 \left\lceil \frac{3}{3} \right\rceil = 2.$$

Table 5: Delay trajectory for $\delta = 5$ at server 1

Cycle	$(p(k), q(k), r(k))$	Description
k_0	$(5, 0, 0)$	Delay starts at server 1
$k_0 + 1$	$(2, 0, 1)$	Delay is absorbed at server 1 while a delay emerges at server 3
$k_0 + 2$	$(1, 0, 0)$	Delay decreases at server 1 while server 3 returns to nominal operation
$k_0 + 3$	$(0, 0, 0)$	System returns to nominal operation

For server 1, the selected delay is within the tolerance threshold $\Delta_1^* = 7$. The system recovers naturally after three cycles, while the corrected upper bound gives

$$N_1^{\min}(5) = 3 \leq 2 \left\lceil \frac{5}{3} \right\rceil = 4.$$

Thus the theorem provides an upper bound rather than an exact recovery formula in general. The illustration confirms the theoretical distinction between critical and non-critical servers: a positive delay at the critical server becomes persistent, whereas a threshold-admissible delay at a non-critical server is absorbed in finite time.

To test the threshold formulas beyond a single numerical trajectory, the recurrence relations and the original max-plus state equation were evaluated for three admissible parameter sets. Set

Table 6: Parameter sets used for the computational validation

Set	d_1	d_2	d_3	t_1	t_2	t_3	t_4	t_5
A	5	8	3	1	0	2	1	1
B	4	9	2	1	1	2	1	1
C	6	10	4	2	0	1	2	1

A is the parameter set used in the preceding illustration, whereas Sets B and C provide two additional configurations with different processing and transfer times. All three sets satisfy the strict unique-critical-self-loop condition in Eq. (14). The parameter values are listed in Table 6.

The margins and delay thresholds derived from these parameter sets are summarized in Table 7; this table also records the largest competing circuit mean used to verify the canonical regime. The quantities are computed from the same definitions used in the analytical derivation. Their values provide the parameter-specific thresholds against which the subsequent boundary tests are compared.

Table 7: Derived margins and thresholds for the validation sets

Set	$\max_{c\neq c_2} \mu(c)$	s_{13}	s_{32}	τ_{d_2}	Δ_1^*	Δ_3^*
A	$6.5 < 8$	4	3	3	7	3
B	$7 < 9$	8	4	5	12	4
C	$8.5 < 10$	6	3	4	9	3

Here $\max_{c\neq c_2} \mu(c)$ denotes the largest mean weight among all elementary circuits other than c_2; hence the strict inequalities in Table 7 independently verify that c_2 is the unique critical circuit in all three cases. We use the notation $\mathrm{R}(n)$ for recovery to the nominal trajectory after n cycles and P for permanent propagation, distinguishing the latter from the permutation matrix P used in the symmetry argument. Tables 8 and 9 test a delay just below the threshold, exactly at the threshold, and just above the threshold.

Table 8: Boundary validation for delays introduced at server 1

Set	Δ_1^*	$\delta = \Delta_1^* - 1$	$\delta = \Delta_1^*$	$\delta = \Delta_1^* + 1$
A	7	R(3)	R(3)	P
B	12	R(3)	R(3)	P
C	9	R(3)	R(3)	P

Table 9: Boundary validation for delays introduced at server 3

Set	Δ_3^*	$\delta = \Delta_3^* - 1$	$\delta = \Delta_3^*$	$\delta = \Delta_3^* + 1$
A	3	R(2)	R(2)	P
B	4	R(2)	R(2)	P
C	3	R(2)	R(2)	P

The boundary transition is particularly transparent for Set A. At server 3,

$$(0,0,2) \to (2,0,0) \to (0,0,0), \qquad (0,0,3) \to (3,0,0) \to (0,0,0),$$

whereas the first super-threshold delay gives

$$(0,0,4) \to (4,1,0) \to (1,1,1) \to (1,1,1) \to \cdots .$$

Thus $\delta = 3$ is still recoverable, while $\delta = 4$ creates a positive critical-server delay and becomes permanent. Similarly, at server 1,

$$(7,0,0) \to (4,0,3) \to (3,0,0) \to (0,0,0),$$

while

$$(8,0,0) \to (5,0,4) \to (4,1,1) \to (1,1,1) \to \cdots.$$

These trajectories directly verify the inclusiveness of the threshold values $\delta = \Delta_i^*$ and the propagative behavior immediately above them. A $\delta = 1$ delay introduced at the critical server 2 remains propagative in Sets A, B, and C, consistently with $\Delta_2^* = 0$.

A second validation was performed by iterating the original max-plus matrix rather than only the reduced delay recurrence. For Set A, the nominal system was initialized at three different eigenvector shifts s_0 and the same threshold delays were introduced at different cycles k_0. The resulting delay vectors were identical in all cases, as summarized in Table 10. This confirms computationally that, within the stated eigenvector-regime scope, the threshold behavior is invariant under max-plus translation of the eigenvector and under the choice of delay cycle.

Table 10: Direct max-plus validation for different eigenvector shifts and delay cycles in Set A

Initial shift s_0	Delay cycle k_0	Server 1, $\delta = 7$	Server 3, $\delta = 3$
0	0	R(3)	R(2)
5	4	R(3)	R(2)
12	9	R(3)	R(2)

For each row of Table 10, direct max-plus iteration yields exactly the same relative-delay sequences

$$(7,0,0) \to (4,0,3) \to (3,0,0) \to (0,0,0)$$

for server 1 and

$$(0,0,3) \to (3,0,0) \to (0,0,0)$$

for server 3. The agreement of these sequences across the tested shifts and delay cycles is consistent with the eigenvector-regime invariance established analytically. Consequently, the expanded computations support all three aspects required by the theoretical results: the exact threshold boundary, robustness across admissible parameter sets, and independence from k_0 and the eigenvector translation within the nominal steady regime.

3.6. Discussion

The max-plus eigenvalue determines the critical circuit and the periodic nominal regime on which the present threshold analysis is based [24]. In the canonical unique-critical-self-loop configuration treated here, every positive additive delay introduced at the critical server persists, whereas delays introduced at the two non-critical servers are naturally recoverable when they do not exceed the derived thresholds. The recovery mechanism is quantified by the two-cycle contraction inequality (30): while the critical component remains undelayed, the maximum delay magnitude decreases by at least τ_{d_2} after each pair of cycles. Consequently, recovery is guaranteed within $2\lceil \delta/\tau_{d_2} \rceil$ cycles. This result is a sufficient upper bound, not an exact recovery time for every parameter set and delay magnitude.

The results can be interpreted as a localized delay-tolerance analysis within the broader literature on robustness of max-plus systems. In global and q-step robustness studies, bounded perturbations are typically introduced at the level of system parameters, and robustness is characterized by the invariance of all or selected state components, either immediately or after finitely many steps [15–18]. The setting considered here is different in that the system matrix is kept unchanged: the perturbation is applied once, directly to a single server event time, and the analysis concerns whether this localized timing disturbance can produce a positive delay on the critical-server component. Accordingly, the critical-circuit margins s_{32} and s_{13} should not be interpreted as parameter-perturbation bounds in the conventional robustness sense. For the canonical model considered in this work, they instead provide exact thresholds for the magnitude of an event-time delay that can be absorbed without inducing a positive delay on the critical server.

A related distinction arises in comparison with switching and feedback-based approaches. Earlier work on the same production setting considered processing-time acceleration as a switching intervention following delay propagation [11], while other max-plus switching studies employ mode changes, supervisory feedback, or feedback control to meet prescribed performance requirements [21, 22]. No such intervention is introduced in the present analysis. Once the initial delay is imposed, the nominal matrix $\overline{A}$ remains unchanged throughout the evolution. The resulting recovery can therefore be regarded as intrinsic to the system dynamics: "natural recovery" here means that, after a finite number of cycles, the event-time trajectory returns to the nominal trajectory because the non-critical paths possess sufficient intrinsic slack. This interpretation distinguishes the recovery mechanism from control-based compensation or switching interventions.

Within this framework, the contribution is intentionally specific rather than a general robustness or control theorem. The analysis yields exact server-specific delay thresholds for the unique-critical-self-loop class and establishes finite-cycle recovery in the eigenvector regime. The extended computations over three parameter sets reproduce the predicted below-threshold, threshold, and above-threshold transition at both non-critical servers. Moreover, direct iteration of the original max-plus state equation under different eigenvector translations and delay cycles produces the same relative-delay trajectories, providing a computational check of the invariance observed in the steady regime. These results support the interpretation of the critical-circuit margins as intrinsic delay-tolerance thresholds of the canonical system, while also delineating the scope of the conclusions. Other critical-circuit configurations, arbitrary transient initial states, repeated or simultaneous event-time disturbances, and controlled recovery following delay propagation are not covered by the present analysis.

4. Conclusion

This study analyzed delay tolerance thresholds and natural recovery in a three-server fork–join production system modeled over max-plus algebra. The analysis focused on the canonical case in which the self-loop at server 2 is the unique critical circuit. Branch symmetry was used to justify the equivalence of the two parallel fork branches, while delay recurrences were used to characterize how a single delay moves among the three servers.

The results show that every positive delay at the critical server propagates, so $\Delta_2^* = 0$. In contrast, non-critical servers can tolerate delays up to the exact threshold values $\Delta_3^* = s_{32}$ and $\Delta_1^* = s_{13} + s_{32}$. If a delay at a non-critical server does not exceed its threshold, the system returns permanently to its nominal trajectory without switching intervention within at most $2\lceil \delta/\tau_{d_2} \rceil$ cycles, where $\tau_{d_2} = \min\{\sigma_1, \sigma_3, s_{13}\}$. The expanded numerical checks reproduce the predicted threshold transition for three distinct admissible parameter sets and confirm identical relative-delay behavior for different delay cycles and eigenvector translations within that regime.

The principal contribution is therefore not a general perturbation-robustness result, but an exact event-delay characterization for the stated canonical fork–join regime: the system matrix remains unchanged, the delay is localized at one server event, and the derived critical-circuit margins determine whether the disturbance is naturally absorbed or becomes permanently propagative. This separates the present result from bounded parameter-perturbation robustness and from switching-based recovery. Future research may extend the analysis to other critical-circuit configurations, arbitrary transient initial states, multiple or repeated delays, stochastic processing times, and controlled switching policies for systems in which delay propagation has already occurred.

CRediT Authorship Contribution Statement

Dewa Putu Wiadnyana Putra: Conceptualization, Methodology, Formal Analysis, Investigation, Writing–Original Draft Preparation, Funding Acquisition. **Marcellinus Andy Rudhito:** Conceptualization, Supervision, Validation, Writing–Review & Editing.

Declaration of Generative AI and AI-assisted technologies

ChatGPT (OpenAI, GPT-5.6 Sol) was used during manuscript revision to assist with language editing, LaTeX formatting, manuscript organization, and computational cross-checking. The authors independently reviewed and verified all mathematical definitions, derivations, proofs, numerical results, interpretations, and references, and take full responsibility for the final content of the manuscript.

Declaration of Competing Interest

The authors declare no competing interests.

Ethics Approval and Consent to Participate

Ethics approval was not required for this study. The research is entirely theoretical and did not involve human participants or personal data; therefore, consent to participate was not applicable.

Funding and Acknowledgments

The authors gratefully acknowledge the financial support provided by the Institute for Research and Community Service of Universitas Sanata Dharma through the SDGs Special-Themed Research Scheme, Contract No. 009 Panel./LPPM-USD/II/2026.

Data and Code Availability

No external dataset was used in this theoretical study. The numerical validation can be reproduced directly from the parameter values, max-plus state equation, and recurrence formulas reported in the manuscript. The verification script used for the reported boundary and regime-invariance checks is available from the corresponding author, Dewa Putu Wiadnyana Putra (dewa@usd.ac.id), upon reasonable request.

References

[1] Robert Ojstersek, Miran Brezocnik, and Borut Buchmeister. "Multi-Objective Optimization of Production Scheduling with Evolutionary Computation: A Review". In: *International Journal of Industrial Engineering Computations* 11.3 (2020), pp. 359–376. DOI: 10.5267/j.ijiec.2020.1.003.

[2] Andrea Rosova, Marcel Behun, Samer Khouri, Michal Cehlar, Viktor Ferencz, and Marian Sofranko. "Case Study: The Simulation Modeling to Improve the Efficiency and Performance of Production Process". In: *Wireless Networks* 28.2 (2022), pp. 863–872. DOI: 10.1007/s11276-020-02341-z.

[3] François Baccelli, Guy Cohen, Geert Jan Olsder, and Jean-Pierre Quadrat. *Synchronization and Linearity: An Algebra for Discrete Event Systems*. Wiley, 1992.

[4] Peter Butkovič. *Max-Linear Systems: Theory and Algorithms*. London: Springer, 2010. DOI: 10.1007/978-1-84996-299-5.

[5] Raymond A. Cuninghame-Green. *Minimax Algebra*. Vol. 166. Lecture Notes in Economics and Mathematical Systems. Springer, 1979.

[6] Bernd Heidergott, Geert Jan Olsder, and Jacob van der Woude. *Max Plus at Work: Modeling and Analysis of Synchronized Systems*. Princeton University Press, 2006.

[7] Marianne Akian, Stéphane Gaubert, and Cormac Walsh. "Discrete Max-Plus Spectral Theory". In: *Idempotent Mathematics and Mathematical Physics*. Ed. by Grigori L. Litvinov and Victor P. Maslov. Vol. 377. Contemporary Mathematics. American Mathematical Society, 2005, pp. 53–77. DOI: 10.1090/conm/377/6982.

[8] Tadao Murata. "Petri Nets: Properties, Analysis and Applications". In: *Proceedings of the IEEE* 77.4 (1989), pp. 541–580. DOI: 10.1109/5.24143.

[9] Cristiano A. V. Cavalcante, A. C. J. Santos, R. G. N. Paiva, A. A. Aribisala, and R. B. da Silva. "Delay Time Model for Determining the Effectiveness of a System under Adverse Conditions of Production and Inspection". In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 238.6 (2024), pp. 1136–1155. DOI: 10.1177/1748006X231207529.

[10] L. Yu, W. Lio, and R. Kang. "Uncertain Delayed Production Risk Process and Its Application in the Shortage for a Production Process System". In: *International Journal of General Systems* 53.7–8 (2024), pp. 1221–1242. DOI: 10.1080/03081079.2024.2358347.

[11] Dewa Putu Wiadnyana Putra and Marcellinus Andy Rudhito. "Analysis and Solution of the Model Production System with Time Delays Using Max-Plus Algebra". In: *International Journal of Mathematics and Computer Research* 12.11 (2024), pp. 4592–4596. DOI: 10.47191/ijmcr/v12i11.07.

[12] Worrawate Leela-apiradee, Phantipa Thipwiwatpotjana, and Artur Gorka. "Closed Form of L-Localized Solution Set of Max-Plus Interval Linear System and Its Application on Optimization Problem". In: *Journal of Computational and Applied Mathematics* 317.1 (2017), pp. 113–127. DOI: 10.1016/j.cam.2016.11.027.

[13] Lihua Wang, Wei Li, and Haohao Li. "AE Solutions to Two-Sided Interval Linear Systems over Max-Plus Algebra". In: *Journal of Inequalities and Applications* 2018.1 (2018), p. 291. DOI: 10.1186/s13660-018-1869-6.

[14] Hongwei Zhang, Yuegang Tao, and Zilong Zhang. "Strong Solvability of Interval Max-Plus Systems and Applications to Optimal Control". In: *Systems & Control Letters* 96.1 (2016), pp. 88–94. DOI: 10.1016/j.sysconle.2016.07.005.

[15] C. Wang and Y. Tao. "Global Robustness for Max-Plus Linear Systems". In: *International Journal of Systems Science* 48.15 (2017), pp. 3225–3232. DOI: 10.1080/00207721.2017.1381890.

[16] Y. Yin, Y. Tao, and C. Wang. "Relatively Maximal Perturbation Bounds for Global Robustness of Max-Plus Linear Systems". In: *International Journal of Robust and Nonlinear Control* 31.9 (2021), pp. 4170–4183. DOI: 10.1002/rnc.5459.

[17] Y. Yin, Y. Tao, C. Wang, and J. Tan. "Global Robustness with q-Step Delay for Max-Plus Linear Systems". In: *IFAC-PapersOnLine* 53.4 (2020), pp. 48–53. DOI: 10.1016/j.ifacol.2021.04.005.

[18] Y. Yin, Y. Tao, C. Wang, and H. Chen. "Local and Global Robustness with q-Step Delay for Max-Plus Linear Systems". In: *Discrete Event Dynamic Systems* 32.2 (2022), pp. 231–251. DOI: 10.1007/s10626-021-00352-2.

[19] A. Gupta, T. van den Boom, J. van der Woude, and B. De Schutter. "Framework for Studying Stability of Switching Max-Plus Linear Systems". In: *IFAC-PapersOnLine* 53.4 (2020), pp. 68–74. DOI: 10.1016/j.ifacol.2021.04.008.

[20] A. Gupta, T. van den Boom, J. van der Woude, and B. De Schutter. "Structural Controllability of Switching Max-Plus Linear Systems". In: *IFAC-PapersOnLine* 53.2 (2020), pp. 1936–1942. DOI: 10.1016/j.ifacol.2020.12.2587.

[21] Ali Mohamadkhani, Marc Geilen, Jeroen Voeten, and Twan Basten. "Modeling and Analysis of Switching Max-Plus Linear Systems with Discrete-Event Feedback". In: *Discrete Event Dynamic Systems* 33.3 (2023), pp. 341–372. DOI: 10.1007/s10626-023-00382-y.

[22] Roohallah Azarmi, Mohsen Alirezaei, Dip Goswami, and Twan Basten. "Tracking Dynamic Deadlines in Switched Max-Plus Linear Systems with Uncontrollable Workloads". In: *Discrete Event Dynamic Systems* 34.4 (2024), pp. 573–604. DOI: 10.1007/s10626-024-00407-0.

[23] Davide Zorzenon, Jan Komenda, and Jörg Raisch. "Switched Max-Plus Linear-Dual Inequalities: Cycle Time Analysis and Applications". In: *Discrete Event Dynamic Systems* 34.1 (2024), pp. 199–250. DOI: 10.1007/s10626-023-00389-5.

[24] H. Al Bermanei, J. M. Böling, and G. Högnäs. "Modeling and Scheduling of Production Systems by Using Max-Plus Algebra". In: *Flexible Services and Manufacturing Journal* 36.1 (2024), pp. 129–150. DOI: 10.1007/s10696-023-09484-z.