Article

Not peer-reviewed version

Weak Monotone Fixed Points for Positive–Negative Guarded Language Systems in a Length-Based Ultrametric Space

Laura Ajeti , Hristo Hristov , Atanas Ilchev , Boyan Zlatanov *

Posted Date: 19 May 2026

doi: 10.20944/preprints202605.1259.v1

Keywords: formal languages; length-based ultrametric spaces; positive-negative guarded systems; weak monotone mappings; ordered contractions; fixed points; Picard iteration; recursive language equations; finite-depth certification; trace-based policies; tool-using AI agents; fixed-point policy semantics

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

Article

Weak Monotone Fixed Points for Positive–Negative Guarded Language Systems in a Length-Based Ultrametric Space

Laura Ajeti [1], Hristo Hristov [2], Atanas Ilchev [3] and and Boyan Zlatanov [3,*]

[1] UBT – University for Business and Technology, Prishtina, Kosovo
[2] Department of Software Technology, Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 4000 Plovdiv, Bulgaria
[3] Department of Mathematical Analysis, Faculty of Mathematics and Informatics, University of Plovdiv Paisii Hilendarski, 4000 Plovdiv, Bulgaria
* Correspondence: bobbyz@uni-plovdiv.bg

Abstract

We study positive–negative guarded systems of language equations over a fixed finite alphabet. The ambient space is the complete ultrametric space of all formal languages equipped with a length-based distance, where two languages are close whenever they agree on all words up to a sufficiently large length. The systems considered here contain both positive recursive dependencies and negative dependencies expressed through language complements. To handle this mixed structure, we introduce a suitable product order on pairs of languages and prove that the associated system operator has the weak monotone property. We show that complement is an isometry for the length-based ultrametric and establish a signed wrapping estimate for guarded positive and negative language terms. These estimates lead to an ordered contraction principle for comparable pairs. As a consequence, the canonical lower and upper Picard iterations converge to the same limit, which is the unique fixed pair of the system. We also derive an explicit convergence rate and a finite-depth certification result: after a prescribed number of iterations, the approximants agree with the fixed-point semantics on all words below a given length. Additional symmetry assumptions are shown to force the unique fixed pair to be diagonal, reducing the system to a single language equation. Finally, we discuss an application to trace-based policies for tool-using AI agents. In this interpretation, finite executions of an agent are represented as words over an alphabet of observable tool-events, and the two components of the fixed point provide a stable semantics for policy-defined admissible and risky trace classes. The resulting framework gives a mathematically certified method for finite-depth analysis of recursive trace-based policies based on ultrametric fixed-point techniques.

Keywords: formal languages; length-based ultrametric spaces; positive-negative guarded systems; weak monotone mappings; ordered contractions; fixed points; Picard iteration; recursive language equations; finite-depth certification; trace-based policies; tool-using AI agents; fixed-point policy semantics

MSC: 47H10; 54E50; 68Q45; 68Q55; 68Q60

---

1. Introduction

The study of fixed point equations on structured spaces is a central theme in both pure and applied mathematics. Since Banach's classical work [1], contraction mappings on complete metric spaces have provided a powerful framework for proving existence and uniqueness of solutions to recursive equations. One of the main advantages of this approach is its constructive nature: the fixed point is obtained as the limit of a Picard iteration sequence, and the contraction constant gives explicit information about the rate of convergence.

 doi:10.20944/preprints202605.1259.v1

Fixed point methods have also proved useful in contexts where recursive or self-referential structures occur naturally. In particular, metric and ultrametric ideas play an important role in the semantics of programming languages and recursive definitions. America and Rutten [2] developed a categorical framework for recursive domain equations in complete metric spaces. A recurring principle in this line of work is that guardedness leads to well-behaved recursion: recursive occurrences become contractive when they are separated from the surrounding context by a sufficient amount of observable structure. In the setting of formal languages, the specific ultrametric used here is related to the Cantor/Bodnarchuk metric structure on language spaces, whose topological background goes back to Vianu [3] and is later discussed by Fulop and Kephart [4].

The present paper develops this philosophy in the setting of formal languages, but with an additional feature that is important for applications: the recursive system may contain both positive and negative dependencies. More precisely, we consider pairs of formal languages over a fixed finite alphabet Σ and study recursive systems in which one component may depend positively on a language and negatively on the complement of another language. Such positive–negative interactions naturally lead to a mixed order structure, because the complement operation reverses inclusion.

The ambient space is the set $\mathcal{L}$ of all formal languages over Σ, equipped with a length-based ultrametric. In this metric, two languages are close when they agree on all words up to a sufficiently large length. Equivalently, the distance is determined by the length of the shortest word on which the two languages differ. This interpretation is particularly suitable for finite-depth approximation: convergence in the ultrametric means stabilization of membership decisions on words of bounded length.

In the present work, we study two-component systems of recursive language equations. Each component is generated from a seed language and a finite family of guarded recursive terms. Some of these terms are positive: they propagate a language through fixed word contexts. Other terms are negative: they involve the complement of a language and therefore express conditions depending on non-membership. The essential assumption is that every recursive occurrence is guarded by a nonempty word context. This guardedness condition is the mechanism that produces contractive behavior in the length-based ultrametric.

The presence of complements creates a genuine order-theoretic difficulty. Purely positive recursive language systems are naturally monotone with respect to ordinary inclusion. In contrast, complement reverses inclusion, and therefore negative occurrences destroy monotonicity in the usual product order. To handle this, we introduce a mixed product order on pairs of languages: the first component is ordered by ordinary inclusion, while the second component is ordered in the reverse direction. With respect to this mixed order, the system operator becomes order-preserving and satisfies the weak monotone property. This allows us to combine order-theoretic arguments with ultrametric contraction estimates.

The main technical ingredients are conceptually simple. First, the complement operation does not change distances in the length-based ultrametric: two languages first disagree at a certain word length if and only if their complements first disagree at the same word length. Second, placing a language, or the complement of a language, inside a fixed word context shifts every possible first disagreement to a larger word length. This is the precise source of the contraction effect produced by guardedness. Third, finite unions of language terms preserve the relevant ultrametric estimates. Together, these facts yield an ordered contraction principle for comparable pairs in the mixed product order.

As a consequence, the canonical lower and upper Picard iterations converge to the same limit. This common limit is the unique fixed pair of the positive–negative guarded language system. Thus, although the system contains negative dependencies through complements, the combination of guardedness, the length-based ultrametric, and the mixed order restores existence, uniqueness, and constructive approximation of the fixed semantics.

The convergence result is also quantitative. The rate of convergence is governed by the minimal guard length: longer guards force possible disagreements to move faster to larger word lengths.

This gives a direct finite-depth interpretation. After sufficiently many iterations, the approximating languages agree with the fixed languages on all words below any prescribed length bound. In particular, for the canonical lower or upper initial approximation, the number of iterations needed to certify all words below a given length is explicitly controlled by that length and by the minimal guard length of the system.

An additional part of the paper studies symmetry. Under a natural symmetry condition, interchanging the two coordinates commutes with the system operator. Since the fixed pair is unique, this forces the fixed pair to be diagonal. In that case, the two-component system reduces to a single fixed language equation on the diagonal. This gives a useful structural criterion for identifying when a positive–negative system has a symmetric equilibrium.

Finally, we discuss an application to the semantics of trace-based policies for tool-using AI agents. In this interpretation, the alphabet consists of abstract observable tool-events, and words represent finite tool-call traces of an agent. The two components of the fixed pair may be interpreted as stable language classes of admissible and risky traces, according to the chosen policy specification. Positive guarded rules propagate admissibility or risk, while negative guarded rules express conditions depending on the absence of membership in the opposite trace class. The finite-depth certification theorem then guarantees that, up to any prescribed trace length, the computed approximation agrees with the unique fixed-point semantics of the policy.

The paper is organized as follows. Section 2 recalls the required preliminaries on formal languages, the length-based ultrametric, complements, and the mixed product order. Section 3 introduces positive–negative guarded language systems and proves the weak monotone and ordered contraction results, together with existence, uniqueness, convergence, symmetry consequences, finite-depth certification, and a concrete symmetric example. Section 4 discusses the application to trace-based policies for tool-using AI agents and explains the finite-depth policy evaluation procedure. Section 5 contains the conclusion and outlines directions for future work.

2. Material and Methods

Throughout, Σ is a fixed finite alphabet, Σ^* is the set of all finite words over Σ, including the empty word ε, and $|w|$ denotes the length of a word $w \in \Sigma^*$.

Definition 1 ([5,6]). *A formal language over Σ is any subset $L \subseteq \Sigma^*$. We denote by $\mathcal{L}$ the set of all formal languages over Σ.*

Definition 2 ([5,6]). *Each language $L \in \mathcal{L}$ is identified with its characteristic function*

$$\mathbf{1}_L : \Sigma^* \to \{0,1\}, \qquad \mathbf{1}_L(w) = \begin{cases} 1, & w \in L, \\ 0, & w \notin L. \end{cases}$$

Definition 3 ([3,4]). *For $L_1, L_2 \in \mathcal{L}$ define*

$$d(L_1, L_2) = \begin{cases} 0, & L_1 = L_2, \\ 2^{-\ell}, & \ell = \min\{|w| : \mathbf{1}_{L_1}(w) \neq \mathbf{1}_{L_2}(w)\}. \end{cases}$$

Theorem 1 ([3,4]). *The ordered pair $(\mathcal{L}, d)$ is a complete metric space and the function $d(\cdot,\cdot)$ a ultrametric.*

Remark 1. *The completeness of $(\mathcal{L}, d)$ can be justified by identifying $\mathcal{L} = \mathcal{P}(\Sigma^*)$ with 2^{Σ^*} via characteristic functions. The metric/topological structure of language spaces of this type goes back to the Bodnarchuk metric space discussed by Vianu [3]. Fulop and Kephart [4] later describe the corresponding Cantor language topology and recall that it is homeomorphic to the Cantor space; in particular, the language space is compact. Since every compact metric space is complete, this provides a standard justification for the completeness of $(\mathcal{L}, d)$.*

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

To simplify the notations we will denote the Cartesian product set $\mathcal{L} \times \mathcal{L}$ with $\mathcal{L}^2$. We introduce the maximum metric $d_\infty : \mathcal{L}^2 \times \mathcal{L}^2 \to [0, +\infty)$ by

$$d_\infty((A_1, A_2), (B_1, B_2)) = \max\{d(A_1, B_1), d(A_2, B_2)\}. \tag{1}$$

Lemma 1. *The space $(\mathcal{L}^2, d_\infty)$ is a complete space and $d_\infty(\cdot, \cdot)$ is a ultrametric..*

The proof follows from Theorem 1 and the fact that the Cartesian product set $\mathcal{L}^2$ endows with the maximum metric is complete metric space. According to [7] the metric d_∞ is an ultrametric.

Definition 4. *For $L \in \mathcal{L}$ define its complement by $L^c = \Sigma^* \setminus L$.*

Definition 5. *For $\sigma \in \{+, -\}$ and $u \in \Sigma^*$ define*

$$u^\sigma = \begin{cases} u, & \sigma = +, \\ u^c, & \sigma = -. \end{cases}$$

Remark 2. *The complement operation reverses inclusion: if $A \subseteq B$, then $B^c \subseteq A^c$. This elementary fact is used repeatedly in the mixed monotone framework.*

The introduction of coupled fixed points in partially ordered metric spaces (X, ρ) is initiated in [8], where if the partially ordered set $(X, \preccurlyeq)$ generates a partial order in the Cartesian product space $(X^2, \preceq)$ is defined as $(x, y) \preceq (u, v)$ if $x \preccurlyeq u$ and $y \succcurlyeq v$. Following this idea, which is widely used in the theory of coupled fixed points in partially ordered metric spaces we introduce a partial order in $\mathcal{L}^2$, where $\mathcal{L}$ is partially ordered by inclusion $\subseteq$.

Definition 6. *We define a partial order on $\mathcal{L}^2$ by $(A_1, A_2) \preceq (B_1, B_2)$, provided that $A_1 \subseteq B_1$ and $B_2 \subseteq A_2$.*

Remark 3. *Thus the first coordinate is ordered by ordinary inclusion, while the second coordinate is ordered by reverse inclusion.*

The investigation of existence of coupled fixed points in partially ordered metric spaces, initiated in [8], needed the considered maps $F : X \times X \to X$ to satisfy the mixed monotone property, i.e., $F(x, z) \preccurlyeq F(u, z)$ and $F(z, y) \succcurlyeq F(z, v)$ for all $x \preccurlyeq u$, $y \preccurlyeq v$ and any $z \in X$. The mixed monotone property means that the map of two variables is an increasing on its first variable and decreasing on it second one.

The idea to investigate maps, sharing monotone property, defined on a partially ordered sets was initiated in [9].

Definition 7 ([9])**.** *Let $(X, \preccurlyeq)$ be a partially ordered set and let $f : X \to X$. We say that f has the weak monotone property if $x \preccurlyeq f(x)$ implies $f(x) \preccurlyeq f^2(x)$, and $x \succcurlyeq f(x)$ implies $f(x) \succcurlyeq f^2(x)$.*

3. Main Results

Definition 8. *Let*

$$G_{11}^+ = \{(u_{11,r}^+, v_{11,r}^+) : r = 1, \dots, p_{11}\}, \quad G_{12}^- = \{(u_{12,r}^-, v_{12,r}^-) : r = 1, \dots, p_{12}\},$$
$$G_{21}^- = \{(u_{21,r}^-, v_{21,r}^-) : r = 1, \dots, p_{21}\}, \quad G_{22}^+ = \{(u_{22,r}^+, v_{22,r}^+) : r = 1, \dots, p_{22}\}$$

be finite, possibly empty, subsets of $\Sigma^* \times \Sigma^*$. The families G_{11}^+ and G_{22}^+ contain positive guards acting on the corresponding variables, while G_{12}^- and G_{21}^- contain negative guards acting on the opposite variables. If one of the numbers $p_{11}, p_{12}, p_{21}, p_{22}$ is equal to 0, then the corresponding inner union is understood as $\emptyset$. We assume

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

$p_{11} + p_{12} + p_{21} + p_{22} \geq 1$. Set $\mathcal{G} = G_{11}^{+} \cup G_{12}^{-} \cup G_{21}^{-} \cup G_{22}^{+}$ and $m = \min\{|u| + |v| : (u, v) \in \mathcal{G}\}$. We assume $m \geq 1$.

Definition 9. *Under the notation of Definition 8, define $T_1, T_2 : \mathcal{L}^2 \to \mathcal{L}$ by*

$$T_1(L_1, L_2) = S_1 \cup \bigcup_{r=1}^{p_{11}} u_{11,r}^{+} L_1 v_{11,r}^{+} \cup \bigcup_{r=1}^{p_{12}} u_{12,r}^{-} L_2^{c} v_{12,r}^{-},$$
$$T_2(L_1, L_2) = S_2 \cup \bigcup_{r=1}^{p_{21}} u_{21,r}^{-} L_1^{c} v_{21,r}^{-} \cup \bigcup_{r=1}^{p_{22}} u_{22,r}^{+} L_2 v_{22,r}^{+}.$$

The corresponding system operator is $\mathbf{T}(L_1, L_2) = \big(T_1(L_1, L_2), T_2(L_1, L_2)\big)$.

Remark 4. *The system from Definition 9 has the standard mixed monotone pattern: the first component is increasing in the first variable and decreasing in the second, while the second component is decreasing in the first variable and increasing in the second. This is exactly the order-theoretic configuration compatible with the mixed product order from Definition 6.*

Lemma 2. *For all $A, B \in \mathcal{L}$, we have $d(A^c, B^c) = d(A, B)$.*

Proof. We distinguish two cases.

Case 1: Let $A = B$: Then $A^c = B^c$, hence $d(A^c, B^c) = 0 = d(A, B)$.

Case 2: Let $A \neq B$: By Definition 3, there exists $\ell = \min\{|w| : \mathbf{1}_A(w) \neq \mathbf{1}_B(w)\}$ such that $d(A, B) = 2^{-\ell}$. For every $w \in \Sigma^*$, we have $\mathbf{1}_{A^c}(w) = 1 - \mathbf{1}_A(w)$ and $\mathbf{1}_{B^c}(w) = 1 - \mathbf{1}_B(w)$. Therefore $\mathbf{1}_{A^c}(w) \neq \mathbf{1}_{B^c}(w)$ if and only if $\mathbf{1}_A(w) \neq \mathbf{1}_B(w)$. Hence A^c and B^c first disagree at the same length ℓ, and so $d(A^c, B^c) = 2^{-\ell} = d(A, B)$. □

Following Definition 5 we present the next notation.

Definition 10. *For $\sigma \in \{+, -\}$ and $L \in \mathcal{L}$ define*

$$L^{\sigma} = \begin{cases} L, & \sigma = +, \\ L^c, & \sigma = -. \end{cases}$$

Lemma 3. *Let $u, v \in \Sigma^*$ and let $\sigma \in \{+, -\}$. Assume $|u| + |v| \geq m \geq 1$. Then, for all $A, B \in \mathcal{L}$, we have $d(uA^{\sigma}v, uB^{\sigma}v) \leq 2^{-m} d(A, B)$.*

Proof. We distinguish two cases according to σ.

Case 1: Let $\sigma =$ "$+$": Then $A^{\sigma} = A$ and $B^{\sigma} = B$. If $A = B$, then $uAv = uBv$, hence $d(uAv, uBv) = 0 = 2^{-m} d(A, B)$, so the inequality is trivial.

Assume now that $A \neq B$, and set $k_0 = \min\{|w| : \mathbf{1}_A(w) \neq \mathbf{1}_B(w)\}$. Then $d(A, B) = 2^{-k_0}$. We prove that uAv and uBv agree on all words of length strictly less than $k_0 + |u| + |v|$. Let $x \in \Sigma^*$ satisfy $|x| < k_0 + |u| + |v|$.

If $x \notin u\Sigma^* v$, then there is no $y \in \Sigma^*$ such that $x = uyv$. Hence $x \notin uAv$ and $x \notin uBv$, so $\mathbf{1}_{uAv}(x) = \mathbf{1}_{uBv}(x) = 0$.

If $x \in u\Sigma^* v$, then $x = uyv$ for some $y \in \Sigma^*$. Since lengths add under concatenation, $|x| = |u| + |y| + |v|$, and therefore $|y| = |x| - |u| - |v| < k_0$. By the definition of k_0, the languages A and B agree on all words of length strictly less than k_0, so $\mathbf{1}_A(y) = \mathbf{1}_B(y)$. Hence $y \in A$ if and only if $y \in B$, and consequently $x = uyv \in uAv$ if and only if $x = uyv \in uBv$. Thus again $\mathbf{1}_{uAv}(x) = \mathbf{1}_{uBv}(x)$.

We have proved that uAv and uBv agree on all words of length strictly less than $k_0 + |u| + |v|$. Therefore their first possible disagreement can occur only at a word of length at least $k_0 + |u| + |v|$, which implies $d(uAv, uBv) \le 2^{-(k_0+|u|+|v|)}$. Since $|u| + |v| \ge m$, we obtain $d(uAv, uBv) \le 2^{-(k_0+m)} = 2^{-m}2^{-k_0} = 2^{-m}d(A, B)$.

Case 2: Let $\sigma = ``-"$: Then $A^\sigma = A^c$ and $B^\sigma = B^c$. Applying the already proved positive case to A^c and B^c, we obtain $d(uA^cv, uB^cv) \le 2^{-m}d(A^c, B^c)$. By Lemma 2, $d(A^c, B^c) = d(A, B)$. Therefore $d(uA^\sigma v, uB^\sigma v) \le 2^{-m}d(A, B)$. □

Lemma 4. *Let I be a finite index set and let $(X_\alpha)_{\alpha\in I}$ and $(Y_\alpha)_{\alpha\in I}$ be families of languages in $\mathcal{L}$. If $d(X_\alpha, Y_\alpha) \le \delta$ for all $\alpha \in I$, then $d\big(\bigcup_{\alpha\in I} X_\alpha, \bigcup_{\alpha\in I} Y_\alpha\big) \le \delta$. In particular, for all $A, B, S \in \mathcal{L}$, we have $d(A \cup S, B \cup S) \le d(A, B)$.*

Proof. Set $U = \bigcup_{\alpha\in I} X_\alpha$ and $V = \bigcup_{\alpha\in I} Y_\alpha$. If $U = V$, then $d(U, V) = 0 \le \delta$. So assume $U \ne V$. Let w_0 be a shortest word such that $\mathbf{1}_U(w_0) \ne \mathbf{1}_V(w_0)$, and set $\ell = |w_0|$. Then $d(U, V) = 2^{-\ell}$.

Without loss of generality, assume $w_0 \in U \setminus V$. Then $w_0 \in X_{\alpha_0}$ for some $\alpha_0 \in I$, while $w_0 \notin Y_{\alpha_0}$. Hence $\mathbf{1}_{X_{\alpha_0}}(w_0) \ne \mathbf{1}_{Y_{\alpha_0}}(w_0)$, so $d(X_{\alpha_0}, Y_{\alpha_0}) \ge 2^{-\ell} = d(U, V)$. Since $d(X_{\alpha_0}, Y_{\alpha_0}) \le \delta$, we conclude that $d(U, V) \le \delta$.

For the special case, take $I = \{1, 2\}$, $X_1 = A$, $X_2 = S$, $Y_1 = B$, and $Y_2 = S$. Then $d(X_1, Y_1) = d(A, B)$ and $d(X_2, Y_2) = d(S, S) = 0 \le d(A, B)$. Applying the general statement with $\delta = d(A, B)$ yields $d(A \cup S, B \cup S) \le d(A, B)$. □

The notion of coupled fixed points for maps of two variable, acting on a Cartesian product space $F : X \times X \to X$, introduced in [10] for normed spaces ordered by a cone. Later, coupled fixed points were investigated in partially ordered metric spaces. Let us recall that an ordered pair $(x, y) \in X^2$ is a coupled fixed point for the map $F : X^2 \to X$ if there hold $x = F(x, y)$ and $y = F(y, x)$. It seems natural, as obtained in most investigations, that a coupled fixed point (x, y) satisfies $x = y$, because of the symmetry $x = F(x, y)$ and $y = F(y, x)$. This drawback was over-passed in [11], by introducing two maps $F_i : X^2 \to X$ and putting the ordered pair $(x, y) \in X^2$ to be a coupled fixed point for the ordered pair of maps (F_1, F_2) if there hold $x = F_1(x, y)$ and $y = F_2(x, y)$. In the particular case when $F_2(x, y) = F_1(y, x)$ we get the classical notion for coupled fixed points introduced in [8,10]. Finally the mixed monotone property for the map $F : X^2 \to X$ was generalized in [12,13].

Definition 11 ([13]). *Let $(X, \preccurlyeq)$ be a partially ordered set and $F_1, F_2 : X \times X \to X$ be two maps. We say that the ordered pair of maps (F_1, F_2) has the mixed monotone property if for any $x, y \in X$ there holds*

$$x_1, x_2 \in X \text{ if } x_1 \preccurlyeq x_2 \text{ then } F_1(x_1, y) \preccurlyeq F_1(x_2, y), F_2(x_1, y) \preccurlyeq F_2(x_2, y)$$

and

$$y_1, y_2 \in X \text{ if } y_1 \preccurlyeq y_2 \text{ then } F_2(x, y_1) \succcurlyeq F_2(x, y_2), F_1(x, y_1) \succcurlyeq F_1(x, y_2)$$

If $F_2(x, y) = F_1(y, x)$ we get the notion of the mixed monotone property from [8].

In what follows we will consider the map the Cartesian product space $\mathcal{L}^2$ and use the notation $\mathbf{L} = (L_1, L_2) \in \mathcal{L}^2$. We will use the partial order, by inclusion, $(\mathcal{L}, \subseteq)$ and partially order $(\mathcal{L}^2, \preceq)$ as $(x, y) \preceq (u, v)$ if $x \subseteq u$ and $y \supseteq v$.

Lemma 5. *Let $\mathbf{L}^{(n)} = (L_1^{(n)}, L_2^{(n)}) \in \mathcal{L}^2$ converge in $(\mathcal{L}^2, d_\infty)$ to $\mathbf{L} = (L_1, L_2)$. If $\mathbf{L}^{(n)} \preceq \mathbf{L}^{(n+1)}$ for all $n \ge 0$, then $\mathbf{L}^{(n)} \preceq \mathbf{L}$ for all $n \ge 0$. If $\mathbf{L}^{(n)} \succeq \mathbf{L}^{(n+1)}$ for all $n \ge 0$, then $\mathbf{L}^{(n)} \succeq \mathbf{L}$ for all $n \ge 0$.*

Proof. We prove the increasing case; the decreasing case is analogous. Assume $\mathbf{L}^{(n)} \preceq \mathbf{L}^{(n+1)}$ for all $n \ge 0$. Then, by Definition 6, $L_1^{(n)} \subseteq L_1^{(n+1)}$ and $L_2^{(n+1)} \subseteq L_2^{(n)}$ for all $n \ge 0$. Fix $n \ge 0$.

We first prove that $L_1^{(n)} \subseteq L_1$. Take any word $w \in L_1^{(n)}$. Since $(L_1^{(m)})_{m\ge n}$ is increasing, we have $w \in L_1^{(m)}$ for all $m \ge n$. If $w \notin L_1$, then for all $m \ge n$ the languages $L_1^{(m)}$ and L_1 differ at the word w,

 doi:10.20944/preprints202605.1259.v1

and so $d(L_1^{(m)}, L_1) \geq 2^{-|w|} > 0$. This contradicts $L_1^{(m)} \to L_1$ in $(\mathcal{L}, d)$. Hence $w \in L_1$, and therefore $L_1^{(n)} \subseteq L_1$.

Next we prove that $L_2 \subseteq L_2^{(n)}$. Take any word $w \in L_2$. Suppose, for contradiction, that $w \notin L_2^{(n)}$. Since $(L_2^{(m)})_{m \geq n}$ is decreasing, we then have $w \notin L_2^{(m)}$ for all $m \geq n$. Thus $L_2^{(m)}$ and L_2 differ at the word w for all $m \geq n$, so $d(L_2^{(m)}, L_2) \geq 2^{-|w|} > 0$, contrary to $L_2^{(m)} \to L_2$. Hence $w \in L_2^{(n)}$, which proves $L_2 \subseteq L_2^{(n)}$.

Thus $L_1^{(n)} \subseteq L_1$ and $L_2 \subseteq L_2^{(n)}$, that is, $\mathbf{L}^{(n)} \preceq \mathbf{L}$. □

In what follows we will consider the map $mathbf{T} = (T_1, T_2) : \mathcal{L}^2 \to \mathcal{L}^2$, where T_1 and T_2 are from Definition 9.

It seems that the ideas from [8,9] can be unified, as suggested in [13] to consider the map $\mathbf{T}$ acting on the partially ordered space $(\mathcal{L}^2, \preceq)$ and to show that $\mathbf{T}$ has the weak monotone property from Definition 7.

Proposition 1. *Let $\mathbf{T} : \mathcal{L}^2 \to \mathcal{L}^2$ be the system operator from Definition 9. Assume that there exists $\mathbf{L}^{(0)} = (L_1^{(0)}, L_2^{(0)}) \in \mathcal{L}^2$ such that either $\mathbf{L}^{(0)} \preceq \mathbf{T}(\mathbf{L}^{(0)})$ or $\mathbf{L}^{(0)} \succeq \mathbf{T}(\mathbf{L}^{(0)})$. Then the Picard sequence $\mathbf{L}^{(n+1)} = \mathbf{T}(\mathbf{L}^{(n)})$, $n \geq 0$, is monotone with respect to $\preceq$. Moreover, $\mathbf{T}$ has the weak monotone property on $(\mathcal{L}^2, \preceq)$.*

Proof. We first prove that $\mathbf{T}$ is order-preserving with respect to the mixed order. Take $(A_1, A_2) \preceq (B_1, B_2)$. Then $A_1 \subseteq B_1$ and $B_2 \subseteq A_2$.

We compare the first component. Let $x \in T_1(A_1, A_2)$. Then one of the following three possibilities holds: $x \in S_1$, $x \in u_{11,r}^{+} A_1 v_{11,r}^{+}$ for some r, or $x \in u_{12,r}^{-} A_2^c v_{12,r}^{-}$ for some r.

If $x \in S_1$, then clearly $x \in T_1(B_1, B_2)$. If $x \in u_{11,r}^{+} A_1 v_{11,r}^{+}$, then $x = u_{11,r}^{+} y v_{11,r}^{+}$ for some $y \in A_1$. Since $A_1 \subseteq B_1$, we have $y \in B_1$, hence $x \in u_{11,r}^{+} B_1 v_{11,r}^{+} \subseteq T_1(B_1, B_2)$. If $x \in u_{12,r}^{-} A_2^c v_{12,r}^{-}$, then $x = u_{12,r}^{-} y v_{12,r}^{-}$ for some $y \in A_2^c$. Since $B_2 \subseteq A_2$, taking complements gives $A_2^c \subseteq B_2^c$. Hence $y \in B_2^c$, so $x \in u_{12,r}^{-} B_2^c v_{12,r}^{-} \subseteq T_1(B_1, B_2)$. Thus $T_1(A_1, A_2) \subseteq T_1(B_1, B_2)$.

Now we compare the second component. We prove that $T_2(B_1, B_2) \subseteq T_2(A_1, A_2)$. Let $x \in T_2(B_1, B_2)$. Then one of the following three possibilities holds: $x \in S_2$, $x \in u_{21,r}^{-} B_1^c v_{21,r}^{-}$ for some r, or $x \in u_{22,r}^{+} B_2 v_{22,r}^{+}$ for some r.

If $x \in S_2$, then clearly $x \in T_2(A_1, A_2)$. If $x \in u_{21,r}^{-} B_1^c v_{21,r}^{-}$, then $x = u_{21,r}^{-} y v_{21,r}^{-}$ for some $y \in B_1^c$. Since $A_1 \subseteq B_1$, taking complements gives $B_1^c \subseteq A_1^c$. Hence $y \in A_1^c$, so $x \in u_{21,r}^{-} A_1^c v_{21,r}^{-} \subseteq T_2(A_1, A_2)$. If $x \in u_{22,r}^{+} B_2 v_{22,r}^{+}$, then $x = u_{22,r}^{+} y v_{22,r}^{+}$ for some $y \in B_2$. Since $B_2 \subseteq A_2$, we have $y \in A_2$, hence $x \in u_{22,r}^{+} A_2 v_{22,r}^{+} \subseteq T_2(A_1, A_2)$. Thus $T_2(B_1, B_2) \subseteq T_2(A_1, A_2)$.

Combining the two inclusions, we obtain $\mathbf{T}(A_1, A_2) \preceq \mathbf{T}(B_1, B_2)$. Hence $\mathbf{T}$ is order-preserving. Now assume that $\mathbf{L}^{(0)} \preceq \mathbf{T}(\mathbf{L}^{(0)}) = \mathbf{L}^{(1)}$. Applying $\mathbf{T}$ to both sides and using order-preservation, we get $\mathbf{L}^{(1)} = \mathbf{T}(\mathbf{L}^{(0)}) \preceq \mathbf{T}(\mathbf{L}^{(1)}) = \mathbf{L}^{(2)}$. Repeating inductively, we obtain $\mathbf{L}^{(n)} \preceq \mathbf{L}^{(n+1)}$ for all $n \geq 0$. The decreasing case is analogous.

Finally, the weak monotone property is immediate from order-preservation: if $\mathbf{Z} \preceq \mathbf{T}(\mathbf{Z})$, then $\mathbf{T}(\mathbf{Z}) \preceq \mathbf{T}^2(\mathbf{Z})$, and similarly for the reversed order. □

Corollary 1. *The system from Definition 9 is genuinely mixed monotone whenever at least one guard occurs in each of the four families G_{11}^{+}, G_{12}^{-}, G_{21}^{-}, and G_{22}^{+}.*

Proof. The first component T_1 depends positively on the first variable and negatively on the second, while the second component T_2 depends negatively on the first variable and positively on the second. If all four families are nonempty, then all four monotonicity directions are effectively present. □

Lemma 6. *If $(A_1, A_2) \preceq (B_1, B_2)$, then, for each $i \in \{1, 2\}$, we have $d\big(T_i(A_1, A_2), T_i(B_1, B_2)\big) \leq 2^{-m} d_\infty\big((A_1, A_2), (B_1, B_2)\big)$. The same conclusion holds if $(A_1, A_2) \succeq (B_1, B_2)$.*

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

Proof. We prove the estimate for the case $(A_1, A_2) \preceq (B_1, B_2)$; the reversed case follows by symmetry of the metric. Set $D = d_\infty((A_1, A_2), (B_1, B_2))$. Then $d(A_1, B_1) \le D$ and $d(A_2, B_2) \le D$.

We first treat the first component. For each positive guard in G_{11}^+, Lemma 3 gives $d(u_{11,r}^+ A_1 v_{11,r}^+, u_{11,r}^+ B_1 v_{11,r}^+) \le 2^{-m} d(A_1, B_1) \le 2^{-m} D$. For each negative guard in G_{12}^-, Lemma 3 gives $d\big(u_{12,r}^- A_2^c v_{12,r}^-, u_{12,r}^- B_2^c v_{12,r}^-\big) \le 2^{-m} d(A_2, B_2) \le 2^{-m} D$. Applying Lemma 4 to all wrapped terms and then adding the common seed S_1, we obtain $d\big(T_1(A_1, A_2), T_1(B_1, B_2)\big) \le 2^{-m} D$.

Now we treat the second component. For each negative guard in G_{21}^-, Lemma 3 gives $d\big(u_{21,r}^- A_1^c v_{21,r}^-, u_{21,r}^- B_1^c v_{21,r}^-\big) \le 2^{-m} d(A_1, B_1) \le 2^{-m} D$. For each positive guard in G_{22}^+, Lemma 3 gives $d\big(u_{22,r}^+ A_2 v_{22,r}^+, u_{22,r}^+ B_2 v_{22,r}^+\big) \le 2^{-m} d(A_2, B_2) \le 2^{-m} D$. Applying Lemma 4 again and adding the common seed S_2, we obtain $d\big(T_2(A_1, A_2), T_2(B_1, B_2)\big) \le 2^{-m} D$.

Thus, for each $i \in \{1, 2\}$, $d\big(T_i(A_1, A_2), T_i(B_1, B_2)\big) \le 2^{-m} D$. If $(A_1, A_2) \succeq (B_1, B_2)$, then equivalently $(B_1, B_2) \preceq (A_1, A_2)$. Applying the already proved case with the roles of the two pairs interchanged and using symmetry of the metric d, we obtain the same conclusion. □

Lemma 7. *If $(A_1, A_2) \preceq (B_1, B_2)$, then*

$$d_\infty\big(\mathbf{T}(A_1, A_2), \mathbf{T}(B_1, B_2)\big) \le 2^{-m} d_\infty\big((A_1, A_2), (B_1, B_2)\big).$$

The same conclusion holds if $(A_1, A_2) \succeq (B_1, B_2)$.

Proof. By Definition 9 and (1) we get

$$d_\infty\big(\mathbf{T}(A_1, A_2), \mathbf{T}(B_1, B_2)\big) = \max\Big\{ d\big(T_1(A_1, A_2), T_1(B_1, B_2)\big), d\big(T_2(A_1, A_2), T_2(B_1, B_2)\big) \Big\}.$$

By Lemma 6, each term inside the maximum is bounded by

$$2^{-m}\, d_\infty\big((A_1, A_2), (B_1, B_2)\big).$$

Therefore the maximum is bounded by the same number. The reversed-order case follows in the same way. □

Theorem 2. *Let $\mathbf{T} : \mathcal{L}^2 \to \mathcal{L}^2$ be the system operator from Definition 9. Define the canonical lower and upper initial pairs by $\mathbf{X}^{(0)} = (\emptyset, \Sigma^*)$ and $\mathbf{Y}^{(0)} = (\Sigma^*, \emptyset)$, and let $\mathbf{X}^{(n+1)} = \mathbf{T}(\mathbf{X}^{(n)})$ and $\mathbf{Y}^{(n+1)} = \mathbf{T}(\mathbf{Y}^{(n)})$, $n \ge 0$.*

Then:

(i) *$(\mathbf{X}^{(n)})_{n \ge 0}$ is increasing with respect to $\preceq$;*
(ii) *$(\mathbf{Y}^{(n)})_{n \ge 0}$ is decreasing with respect to $\preceq$;*
(iii) *$\mathbf{X}^{(n)} \preceq \mathbf{Y}^{(n)}$ for every $n \ge 0$;*
(iv) *both sequences converge in $(\mathcal{L}^2, d_\infty)$ to the same limit $\mathbf{L}^* = (L_1^*, L_2^*) \in \mathcal{L}^2$;*
(v) *$\mathbf{L}^*$ is a fixed pair of $\mathbf{T}$;*
(vi) *$\mathbf{L}^*$ is the unique fixed pair of $\mathbf{T}$.*

Proof. We first verify that the lower initial pair is below its image. By definition of the mixed order, $\mathbf{X}^{(0)} \preceq \mathbf{X}^{(1)}$ means $\emptyset \subseteq T_1(\emptyset, \Sigma^*)$ and $T_2(\emptyset, \Sigma^*) \subseteq \Sigma^*$. Both statements are immediate: the first holds because $\emptyset$ is contained in every language, and the second holds because $T_2(\emptyset, \Sigma^*) \subseteq \Sigma^*$ by construction.

Similarly, $\mathbf{Y}^{(1)} \preceq \mathbf{Y}^{(0)}$ means $T_1(\Sigma^*, \emptyset) \subseteq \Sigma^*$ and $\emptyset \subseteq T_2(\Sigma^*, \emptyset)$, and both inclusions are again immediate.

By Proposition 1, it follows that $\mathbf{X}^{(n)} \preceq \mathbf{X}^{(n+1)}$ and $\mathbf{Y}^{(n+1)} \preceq \mathbf{Y}^{(n)}$ for all $n \ge 0$. Thus $(\mathbf{X}^{(n)})$ is increasing and $(\mathbf{Y}^{(n)})$ is decreasing.

Next, we show that $\mathbf{X}^{(n)} \preceq \mathbf{Y}^{(n)}$ for all $n \ge 0$. For $n = 0$ this is clear, because $\emptyset \subseteq \Sigma^*$ in both coordinates, so $(\emptyset, \Sigma^*) \preceq (\Sigma^*, \emptyset)$. Assume now that $\mathbf{X}^{(n)} \preceq \mathbf{Y}^{(n)}$. Since $\mathbf{T}$ is order-preserving by Proposition 1, we obtain $\mathbf{X}^{(n+1)} = \mathbf{T}(\mathbf{X}^{(n)}) \preceq \mathbf{T}(\mathbf{Y}^{(n)}) = \mathbf{Y}^{(n+1)}$. Thus the claim follows by induction.

 doi:10.20944/preprints202605.1259.v1

We now prove that both sequences are Cauchy. By Lemma 7, for every $n \geq 1$,

$$d_\infty(\mathbf{X}^{(n+1)}, \mathbf{X}^{(n)}) \leq 2^{-m} d_\infty(\mathbf{X}^{(n)}, \mathbf{X}^{(n-1)}).$$

Iterating this estimate gives $d_\infty(\mathbf{X}^{(n+1)}, \mathbf{X}^{(n)}) \leq 2^{-mn} d_\infty(\mathbf{X}^{(1)}, \mathbf{X}^{(0)})$.
Hence, if $p > q$, then by the ordinary triangle inequality,

$$d_\infty(\mathbf{X}^{(p)}, \mathbf{X}^{(q)}) \leq d_\infty(\mathbf{X}^{(1)}, \mathbf{X}^{(0)}) \sum_{k=q}^{p-1} 2^{-mk}.$$

Since $m \geq 1$, we have $0 < 2^{-m} < 1$, and the geometric tail tends to 0 as $q \to \infty$. Therefore $(\mathbf{X}^{(n)})$ is Cauchy in $(\mathcal{L}^2, d_\infty)$.

Exactly the same argument shows that $(\mathbf{Y}^{(n)})$ is Cauchy. Since $(\mathcal{L}^2, d_\infty)$ is complete by Lemma 1, there exist $\mathbf{X}^*, \mathbf{Y}^* \in \mathcal{L}^2$ such that $\mathbf{X}^{(n)} \to \mathbf{X}^*$ and $\mathbf{Y}^{(n)} \to \mathbf{Y}^*$.

We show next that both limits are fixed pairs. Since $(\mathbf{X}^{(n)})$ is increasing, Lemma 5 yields $\mathbf{X}^{(n)} \preceq \mathbf{X}^*$ for all $n \geq 0$. Hence every pair $(\mathbf{X}^{(n)}, \mathbf{X}^*)$ is comparable. Therefore, by Lemma 7, $d_\infty(\mathbf{T}(\mathbf{X}^{(n)}), \mathbf{T}(\mathbf{X}^*)) \leq 2^{-m} d_\infty(\mathbf{X}^{(n)}, \mathbf{X}^*)$.

Using $\mathbf{X}^{(n+1)} = \mathbf{T}(\mathbf{X}^{(n)})$, we obtain

$$\begin{aligned} d_\infty(\mathbf{X}^*, \mathbf{T}(\mathbf{X}^*)) &\leq d_\infty(\mathbf{X}^*, \mathbf{X}^{(n+1)}) + d_\infty(\mathbf{X}^{(n+1)}, \mathbf{T}(\mathbf{X}^*)) \\ &= d_\infty(\mathbf{X}^*, \mathbf{X}^{(n+1)}) + d_\infty(\mathbf{T}(\mathbf{X}^{(n)}), \mathbf{T}(\mathbf{X}^*)) \\ &\leq d_\infty(\mathbf{X}^*, \mathbf{X}^{(n+1)}) + 2^{-m} d_\infty(\mathbf{X}^{(n)}, \mathbf{X}^*). \end{aligned}$$

Letting $n \to \infty$, both terms on the right-hand side tend to 0, hence $d_\infty(\mathbf{X}^*, \mathbf{T}(\mathbf{X}^*)) = 0$. Therefore $\mathbf{T}(\mathbf{X}^*) = \mathbf{X}^*$, so $\mathbf{X}^*$ is a fixed pair. The proof that $\mathbf{Y}^*$ is a fixed pair is identical.

We now prove that the two limits coincide. Since $\mathbf{X}^{(n)} \preceq \mathbf{Y}^{(n)}$ for all $n \geq 0$, Lemma 7 gives $d_\infty(\mathbf{X}^{(n+1)}, \mathbf{Y}^{(n+1)}) \leq 2^{-m} d_\infty(\mathbf{X}^{(n)}, \mathbf{Y}^{(n)})$.

Iterating, $d_\infty(\mathbf{X}^{(n)}, \mathbf{Y}^{(n)}) \leq 2^{-mn} d_\infty(\mathbf{X}^{(0)}, \mathbf{Y}^{(0)})$, and hence $d_\infty(\mathbf{X}^{(n)}, \mathbf{Y}^{(n)}) \to 0$. Using the ordinary triangle inequality,

$$d_\infty(\mathbf{X}^*, \mathbf{Y}^*) \leq d_\infty(\mathbf{X}^*, \mathbf{X}^{(n)}) + d_\infty(\mathbf{X}^{(n)}, \mathbf{Y}^{(n)}) + d_\infty(\mathbf{Y}^{(n)}, \mathbf{Y}^*).$$

Letting $n \to \infty$, the right-hand side tends to 0. Therefore $d_\infty(\mathbf{X}^*, \mathbf{Y}^*) = 0$, so $\mathbf{X}^* = \mathbf{Y}^*$. Denote this common limit by $\mathbf{L}^* = (L_1^*, L_2^*)$.

Finally, we prove uniqueness. Let $\mathbf{Z} = (Z_1, Z_2) \in \mathcal{L}^2$ be any fixed pair of $\mathbf{T}$. Since $\emptyset \subseteq Z_1 \subseteq \Sigma^*$ and $\emptyset \subseteq Z_2 \subseteq \Sigma^*$, we have $(\emptyset, \Sigma^*) \preceq (Z_1, Z_2) \preceq (\Sigma^*, \emptyset)$, that is, $\mathbf{X}^{(0)} \preceq \mathbf{Z} \preceq \mathbf{Y}^{(0)}$. Because $\mathbf{T}$ is order-preserving and $\mathbf{T}(\mathbf{Z}) = \mathbf{Z}$, it follows inductively that $\mathbf{X}^{(n)} \preceq \mathbf{Z} \preceq \mathbf{Y}^{(n)}$ for all $n \geq 0$.

Now $d_\infty(\mathbf{X}^{(n+1)}, \mathbf{Z}) = d_\infty(\mathbf{T}(\mathbf{X}^{(n)}), \mathbf{T}(\mathbf{Z})) \leq 2^{-m} d_\infty(\mathbf{X}^{(n)}, \mathbf{Z})$, and hence

$$\lim_{n \to \infty} d_\infty(\mathbf{X}^{(n)}, \mathbf{Z}) = 0.$$

But also $\lim_{n \to \infty} \mathbf{X}^{(n)} = \mathbf{L}^*$. Therefore $\mathbf{Z} = \mathbf{L}^*$, so the fixed pair is unique. □

Corollary 2. *Under the assumptions of Definitions 8 and 9, let $\mathbf{L}^{(0)} = (L_1^{(0)}, L_2^{(0)}) \in \mathcal{L}^2$ be any initial pair such that either $\mathbf{L}^{(0)} \preceq \mathbf{T}(\mathbf{L}^{(0)})$ or $\mathbf{L}^{(0)} \succeq \mathbf{T}(\mathbf{L}^{(0)})$. Then the Picard iteration $\mathbf{L}^{(n+1)} = \mathbf{T}(\mathbf{L}^{(n)})$, $n \geq 0$, converges in $(\mathcal{L}^2, d_\infty)$ to the unique fixed pair $\mathbf{L}^* = (L_1^*, L_2^*)$ from Theorem 2.*

Proof. By Proposition 1, the Picard sequence is monotone. We treat the increasing case $\mathbf{L}^{(n)} \preceq \mathbf{L}^{(n+1)}$ for all $n \geq 0$; the decreasing case is analogous.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

Since consecutive iterates are comparable, Lemma 7 gives

$$d_\infty(\mathbf{L}^{(n+2)}, \mathbf{L}^{(n+1)}) \le 2^{-m} d_\infty(\mathbf{L}^{(n+1)}, \mathbf{L}^{(n)}).$$

Iterating, we get $d_\infty(\mathbf{L}^{(n+1)}, \mathbf{L}^{(n)}) \le 2^{-mn} d_\infty(\mathbf{L}^{(1)}, \mathbf{L}^{(0)})$.

Hence, exactly as in the proof of Theorem 2, the sequence is Cauchy in $(\mathcal{L}^2, d_\infty)$ and therefore converges to some limit $\mathbf{Z}^* \in \mathcal{L}^2$.

Using Lemma 5 and the same fixed-point argument as in Theorem 2, we conclude that $\mathbf{T}(\mathbf{Z}^*) = \mathbf{Z}^*$. Thus $\mathbf{Z}^*$ is a fixed pair of $\mathbf{T}$. By the uniqueness part of Theorem 2, this fixed pair must coincide with the unique fixed pair $\mathbf{L}^* = (L_1^*, L_2^*)$. Therefore $\mathbf{L}^{(n)} \to \mathbf{L}^*$. □

Symmetry Consequences

Definition 12. *Define the coordinate-swap map $P : \mathcal{L}^2 \to \mathcal{L}^2$ by $P(A, B) = (B, A)$. We also define the diagonal subset $\Delta = \{(L, L) : L \in \mathcal{L}\} \subseteq \mathcal{L}^2$.*

Lemma 8. *Assume that $T_2(A, B) = T_1(B, A)$ for all $A, B \in \mathcal{L}$. Then $P \circ \mathbf{T} = \mathbf{T} \circ P$. Moreover, the diagonal set Δ is invariant under $\mathbf{T}$.*

Proof. Take any $(A, B) \in \mathcal{L}^2$. By Definition 9, $\mathbf{T}(A, B) = \big(T_1(A, B), T_2(A, B)\big)$, hence $P(\mathbf{T}(A, B)) = \big(T_2(A, B), T_1(A, B)\big)$. Using the symmetry assumption, $T_2(A, B) = T_1(B, A)$ and, with (B, A) in place of (A, B), $T_2(B, A) = T_1(A, B)$. Therefore

$$P(\mathbf{T}(A, B)) = \big(T_1(B, A), T_2(B, A)\big) = \mathbf{T}(B, A) = \mathbf{T}(P(A, B)).$$

Hence $P \circ \mathbf{T} = \mathbf{T} \circ P$.

Now let $(L, L) \in \Delta$. Then $\mathbf{T}(L, L) = \big(T_1(L, L), T_2(L, L)\big)$. By symmetry, $T_2(L, L) = T_1(L, L)$, so $\mathbf{T}(L, L) = \big(T_1(L, L), T_1(L, L)\big) \in \Delta$. Thus Δ is invariant under $\mathbf{T}$. □

Proposition 2. *Assume that $T_2(A, B) = T_1(B, A)$ for all $A, B \in \mathcal{L}$. If $(L_1^*, L_2^*) \in \mathcal{L}^2$ is a fixed pair of $\mathbf{T}$, then (L_2^*, L_1^*) is also a fixed pair of $\mathbf{T}$.*

Proof. Assume that $\mathbf{T}(L_1^*, L_2^*) = (L_1^*, L_2^*)$. Applying P to both sides gives $P(\mathbf{T}(L_1^*, L_2^*)) = (L_2^*, L_1^*)$. By Lemma 8, $P(\mathbf{T}(L_1^*, L_2^*)) = \mathbf{T}(P(L_1^*, L_2^*)) = \mathbf{T}(L_2^*, L_1^*)$. Therefore $\mathbf{T}(L_2^*, L_1^*) = (L_2^*, L_1^*)$, which proves that (L_2^*, L_1^*) is also a fixed pair. □

Corollary 3. *Assume that $T_2(A, B) = T_1(B, A)$ for all $A, B \in \mathcal{L}$. Then the unique fixed pair (L_1^*, L_2^*) of $\mathbf{T}$ from Theorem 2 satisfies $L_1^* = L_2^*$. Equivalently, $(L_1^*, L_2^*) \in \Delta$.*

Proof. By Theorem 2, the operator $\mathbf{T}$ has a unique fixed pair (L_1^*, L_2^*). By Proposition 2, the swapped pair (L_2^*, L_1^*) is also a fixed pair of $\mathbf{T}$. By uniqueness, the two fixed pairs must coincide, that is, $(L_1^*, L_2^*) = (L_2^*, L_1^*)$. Comparing coordinates gives $L_1^* = L_2^*$. Hence the unique fixed pair is diagonal. □

Corollary 4. *Assume that $T_2(A, B) = T_1(B, A)$ for all $A, B \in \mathcal{L}$. Define $\widetilde{T} : \mathcal{L} \to \mathcal{L}$ by $\widetilde{T}(L) = T_1(L, L)$. Then there exists a unique language $L^* \in \mathcal{L}$ such that $\widetilde{T}(L^*) = L^*$, and the unique fixed pair of $\mathbf{T}$ is exactly (L^*, L^*).*

Proof. By Corollary 3, the unique fixed pair of $\mathbf{T}$ has the form (L^*, L^*) for some $L^* \in \mathcal{L}$. Since $\mathbf{T}(L^*, L^*) = (L^*, L^*)$, we have $T_1(L^*, L^*) = L^*$, and therefore $\widetilde{T}(L^*) = L^*$.

Conversely, let $L \in \mathcal{L}$ satisfy $\widetilde{T}(L) = L$. Then $T_1(L, L) = L$. By the symmetry assumption, $T_2(L, L) = T_1(L, L) = L$. Hence $\mathbf{T}(L, L) = (L, L)$, so (L, L) is a fixed pair of $\mathbf{T}$. By uniqueness of the fixed pair, $(L, L) = (L^*, L^*)$, and therefore $L = L^*$. Thus L^* is the unique fixed point of $\widetilde{T}$. □

 doi:10.20944/preprints202605.1259.v1

Corollary 5. *Assume that $T_2(A,B) = T_1(B,A)$ for all $A,B \in \mathcal{L}$. Assume also that the Picard iteration starts from a diagonal initial pair $\mathbf{L}^{(0)} = (L^{(0)}, L^{(0)})$ and that this initial pair is comparable with its image under $\mathbf{T}$. Then every iterate is diagonal, that is, $\mathbf{L}^{(n)} \in \Delta$ for all $n \geq 0$. If, moreover, the Picard iteration converges to a fixed pair $\mathbf{L}^*$, then $\mathbf{L}^* \in \Delta$.*

Proof. Since $\mathbf{L}^{(0)} \in \Delta$ and Δ is invariant under $\mathbf{T}$ by Lemma 8, it follows inductively that $\mathbf{L}^{(n)} \in \Delta$ for all $n \geq 0$.

Assume now that $\mathbf{L}^{(n)} \to \mathbf{L}^*$ in $(\mathcal{L}^2, d_\infty)$. We show that Δ is closed in $(\mathcal{L}^2, d_\infty)$. Let $\mathbf{L}^{(n)} = (M^{(n)}, M^{(n)})$ for all $n \geq 0$, and assume that $\mathbf{L}^{(n)} \to (A,B)$ in $(\mathcal{L}^2, d_\infty)$. Then $d(M^{(n)}, A) \to 0$ and $d(M^{(n)}, B) \to 0$. By the strong triangle inequality for the ultrametric d, i.e., $d(A,B) \leq \max\{d(A, M^{(n)}), d(M^{(n)}, B)\}$, letting $n \to \infty$, the right-hand side tends to 0, and hence $d(A,B) = 0$. Therefore $A = B$, so $(A,B) \in \Delta$. Thus Δ is closed. Since every iterate belongs to Δ and the sequence converges to $\mathbf{L}^*$, it follows that $\mathbf{L}^* \in \Delta$. □

Remark 5. *The symmetry assumption $T_2(A,B) = T_1(B,A)$ for all $A,B \in \mathcal{L}$ is an additional structural condition, independent of the weak monotone contraction argument. It implies that the fixed-point set is stable under coordinate exchange; under uniqueness, this forces a diagonal equilibrium.*

Lemma 9. *Assume that the hypotheses of Corollary 2 hold, and let $\mathbf{L}^* = (L_1^*, L_2^*)$ be the fixed pair obtained as the limit of the monotone Picard iteration starting from $\mathbf{L}^{(0)} = (L_1^{(0)}, L_2^{(0)})$. Then, for every integer $n \geq 1$ the inequality $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$ holds.*

Proof. By Proposition 1, the Picard sequence is monotone. We treat the increasing case; the decreasing case is analogous. Thus $\mathbf{L}^{(n)} \preceq \mathbf{L}^{(n+1)}$ for all $n \geq 0$. Since $\mathbf{L}^{(n)} \to \mathbf{L}^*$, Lemma 5 yields $\mathbf{L}^{(n)} \preceq \mathbf{L}^*$ for all $n \geq 0$. We prove the estimate by induction on n.

Base step: $n = 1$.
Since $\mathbf{T}(\mathbf{L}^*) = \mathbf{L}^*$, we obtain $d_\infty(\mathbf{L}^{(1)}, \mathbf{L}^*) = d_\infty(\mathbf{T}(\mathbf{L}^{(0)}), \mathbf{T}(\mathbf{L}^*)) \leq 2^{-m} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$.

Induction step.
Assume $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$. Then

$$d_\infty(\mathbf{L}^{(n+1)}, \mathbf{L}^*) = d_\infty(\mathbf{T}(\mathbf{L}^{(n)}), \mathbf{T}(\mathbf{L}^*)) \leq 2^{-m} d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*),$$

and hence $d_\infty(\mathbf{L}^{(n+1)}, \mathbf{L}^*) \leq 2^{-m(n+1)} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$. This completes the induction. □

Corollary 6. *Assume that the hypotheses of Corollary 2 hold, and let $\mathbf{L}^* = (L_1^*, L_2^*)$ be the unique fixed pair of $\mathbf{T}$. Assume also that $d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) > 0$. If we want $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-N}$, it is sufficient to require*

$$n \geq \frac{N - \log_2 d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)}{m}.$$

In particular, if the iteration starts from one of the canonical initial pairs $(\emptyset, \Sigma^*)$ or $(\Sigma^*, \emptyset)$, then it is sufficient to take $n \geq \lceil N/m \rceil$.

Proof. By Lemma 9, $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$. Therefore, in order to guarantee $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-N}$, it is sufficient to require $2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) \leq 2^{-N}$.

Taking base-2 logarithms, we obtain $-mn + \log_2 d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) \leq -N$, and hence

$$n \geq \frac{N - \log_2 d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)}{m}.$$

Now consider the special case of the canonical lower or upper initial pair. Since every value of d_∞ lies in $[0,1]$, we have $d_\infty((\emptyset, \Sigma^*), \mathbf{L}^*) \leq 1$ and $d_\infty((\Sigma^*, \emptyset), \mathbf{L}^*) \leq 1$. Therefore Lemma 9 yields

 doi:10.20944/preprints202605.1259.v1

$d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \le 2^{-mn}$. Hence it is sufficient to require $2^{-mn} \le 2^{-N}$, equivalently $mn \ge N$. Since n is an integer, it is sufficient to take $n \ge \lceil N/m \rceil$. □

Theorem 3. *If $d_\infty((L_1, L_2), (L_1^*, L_2^*)) \le 2^{-N}$, then, for each $i \in \{1, 2\}$, we have $\mathbf{1}_{L_i}(w) = \mathbf{1}_{L_i^*}(w)$ for all $w \in \Sigma^*$ with $|w| < N$.*

Proof. By (1), there holds $d_\infty((L_1, L_2), (L_1^*, L_2^*)) = \max\{d(L_1, L_1^*), d(L_2, L_2^*)\}$. Hence $d(L_i, L_i^*) \le 2^{-N}$ for $i = 1, 2$. Fix i. If $d(L_i, L_i^*) = 0$, then $L_i = L_i^*$, and the conclusion is immediate. If $d(L_i, L_i^*) > 0$, then by Definition 3, $d(L_i, L_i^*) = 2^{-\ell}$ for some $\ell \ge 0$, where ℓ is the smallest length at which the two languages differ. Since $2^{-\ell} \le 2^{-N}$, we must have $\ell \ge N$. Therefore the two languages coincide on all words of length strictly less than N. □

Corollary 7. *Assume that the hypotheses of Corollary 2 hold, and let $\mathbf{L}^* = (L_1^*, L_2^*)$ be the unique fixed pair of $\mathbf{T}$. If $d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) = 0$, then $\mathbf{L}^{(n)} = \mathbf{L}^*$ for all $n \ge 0$, and the conclusion is immediate.*

Assume now that $d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) > 0$. If

$$n \ge \frac{N - \log_2 d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)}{m},$$

then $\mathbf{1}_{L_i^{(n)}}(w) = \mathbf{1}_{L_i^*}(w)$ for all $w \in \Sigma^*$ with $|w| < N$ and $i = 1, 2$.

In particular, if the iteration starts from one of the canonical initial pairs $(\emptyset, \Sigma^*)$ or $(\Sigma^*, \emptyset)$, then $n \ge \lceil N/m \rceil$ is sufficient to guarantee coordinatewise agreement with the fixed pair on all words of length strictly less than N.

Proof. If $d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) = 0$, then $\mathbf{L}^{(0)} = \mathbf{L}^*$. Since $\mathbf{L}^*$ is a fixed pair of $\mathbf{T}$, it follows that $\mathbf{L}^{(n)} = \mathbf{L}^*$ for all $n \ge 0$, so the conclusion is immediate.

Assume now that $d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*) > 0$. By Corollary 6, the stated bound on n implies $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \le 2^{-N}$. Applying Theorem 3, we obtain $\mathbf{1}_{L_i^{(n)}}(w) = \mathbf{1}_{L_i^*}(w)$ for all $w \in \Sigma^*$ with $|w| < N$ and $i = 1, 2$.

The special case of the canonical initial pairs follows from the second part of Corollary 6. □

3.1. Construction of the Fixed Point via Canonical Iteration

We illustrate Theorem 2.13 on a concrete symmetric system and compute the fixed point by means of the canonical lower iteration.

Let the alphabet be $\Sigma = \{a, b\}$, and define

$$T_1(A, B) = \{\varepsilon\} \cup aA \cup bB^c, \qquad T_2(A, B) = \{\varepsilon\} \cup aB \cup bA^c.$$

Then $T_2(A, B) = T_1(B, A)$ for all $A, B \in \mathcal{L}$, so the symmetry assumption of Lemma 8 and Corollaries 3–4 is satisfied. Hence the unique fixed pair of the system is diagonal, i.e., it has the form (L^*, L^*).

We now construct this fixed pair via the canonical lower iteration $X^{(0)} = (\emptyset, \Sigma^*)$ and $X^{(n+1)} = \mathbf{T}(X^{(n)})$, $n \ge 0$.

Step 1: the first few iterates.

Starting from $X^{(0)} = (\emptyset, \Sigma^*)$, we obtain

$$X_1^{(1)} = T_1(\emptyset, \Sigma^*) = \{\varepsilon\} \cup a\emptyset \cup b(\Sigma^*)^c = \{\varepsilon\},$$

because $a\emptyset = \emptyset$ and $(\Sigma^*)^c = \emptyset$. Moreover,

$$X_2^{(1)} = T_2(\emptyset, \Sigma^*) = \{\varepsilon\} \cup a\Sigma^* \cup b\emptyset^c = \{\varepsilon\} \cup a\Sigma^* \cup b\Sigma^* = \Sigma^*.$$

Thus $X^{(1)} = (\{\varepsilon\}, \Sigma^*)$.

 doi:10.20944/preprints202605.1259.v1

Applying $\mathbf{T}$ once more, we get $X^{(2)} = \mathbf{T}(X^{(1)})$. Its first coordinate is

$$X_1^{(2)} = T_1(\{\varepsilon\}, \Sigma^*) = \{\varepsilon\} \cup a\{\varepsilon\} \cup b(\Sigma^*)^c = \{\varepsilon, a\}$$

and its second coordinate is

$$X_2^{(2)} = T_2(\{\varepsilon\}, \Sigma^*) = \{\varepsilon\} \cup a\Sigma^* \cup b(\{\varepsilon\})^c = \Sigma^* \setminus \{b\}.$$

Hence $X^{(2)} = (\{\varepsilon, a\}, \Sigma^* \setminus \{b\})$.

For the third iterate, since $X_1^{(2)} = \{\varepsilon, a\}$ and $(X_2^{(2)})^c = \{b\}$, we get

$$X_1^{(3)} = T_1(X^{(2)}) = \{\varepsilon\} \cup aX_1^{(2)} \cup b(X_2^{(2)})^c = \{\varepsilon, a, aa, bb\}.$$

Thus the first coordinate begins as $X_1^{(1)} = \{\varepsilon\}$, $X_1^{(2)} = \{\varepsilon, a\}$, and $X_1^{(3)} = \{\varepsilon, a, aa, bb\}$. This suggests that $X_1^{(n)}$ consists of the words of length $< n$ having an even number of occurrences of b.

Step 2: the precise inductive description.

For each $n \geq 1$, define $E_n = \{\, w \in \Sigma^* : |w| < n \text{ and } \#_b(w) \text{ is even} \,\}$ and $O_n = \{\, w \in \Sigma^* : |w| < n \text{ and } \#_b(w) \text{ is odd} \,\}$. We claim that, for every $n \geq 1$, $X_1^{(n)} = E_n$ and $(X_2^{(n)})^c = O_n$.

Base case: $n = 1$**.**

From the computation above, $X^{(1)} = (\{\varepsilon\}, \Sigma^*)$. Therefore $X_1^{(1)} = \{\varepsilon\} = E_1$, because ε is the unique word of length < 1, and it contains an even number of occurrences of b. Also, $(X_2^{(1)})^c = (\Sigma^*)^c = \varnothing = O_1$, since there is no word of length < 1 with an odd number of occurrences of b. Thus the claim is true for $n = 1$.

Inductive step.

Assume that, for some $n \geq 1$, we have $X_1^{(n)} = E_n$ and $(X_2^{(n)})^c = O_n$. We prove the corresponding identities for $n + 1$.

First coordinate. Using the definition of T_1, we have $X_1^{(n+1)} = \{\varepsilon\} \cup aX_1^{(n)} \cup b(X_2^{(n)})^c = \{\varepsilon\} \cup aE_n \cup bO_n$. We show that this is exactly E_{n+1}.

Inclusion $\subseteq$. Every word in $\{\varepsilon\}$ clearly has even b-parity and length $< n + 1$. If $w \in aE_n$, then $w = az$ for some $z \in E_n$; hence $|z| < n$, so $|w| < n + 1$, and $\#_b(w) = \#_b(z)$ is even. If $w \in bO_n$, then $w = bz$ for some $z \in O_n$; since z has odd b-parity, the extra initial b makes $\#_b(w)$ even, and again $|w| < n + 1$. Thus every word in $\{\varepsilon\} \cup aE_n \cup bO_n$ belongs to E_{n+1}.

Inclusion $\supseteq$. Let $w \in E_{n+1}$. If $w = \varepsilon$, then $w \in \{\varepsilon\} \subseteq \{\varepsilon\} \cup aE_n \cup bO_n$. Assume now that $w \neq \varepsilon$. Since $\Sigma = \{a, b\}$, the word w begins either with a or with b. If $w = az$, then $|z| < n$ and $\#_b(z) = \#_b(w)$ is even; hence $z \in E_n$, so $w \in aE_n$. If $w = bz$, then $|z| < n$ and $\#_b(z) = \#_b(w) - 1$ is odd; hence $z \in O_n$, so $w \in bO_n$. Thus every word of E_{n+1} belongs to $\{\varepsilon\} \cup aE_n \cup bO_n$. Therefore $X_1^{(n+1)} = E_{n+1}$.

Second coordinate. Using the definition of T_2, we get $X_2^{(n+1)} = \{\varepsilon\} \cup aX_2^{(n)} \cup b(X_1^{(n)})^c$. By the induction hypothesis, this becomes $X_2^{(n+1)} = \{\varepsilon\} \cup a(\Sigma^* \setminus O_n) \cup b(\Sigma^* \setminus E_n)$. We claim that $X_2^{(n+1)} = \Sigma^* \setminus O_{n+1}$, or equivalently, $(X_2^{(n+1)})^c = O_{n+1}$.

Inclusion $X_2^{(n+1)} \subseteq \Sigma^* \setminus O_{n+1}$. Take $w \in X_2^{(n+1)}$. If $w = \varepsilon$, then clearly $w \notin O_{n+1}$. If $w = az$ with $z \in \Sigma^* \setminus O_n$, then either $|z| \geq n$, hence $|w| \geq n + 1$, or $|z| < n$ and $\#_b(z)$ is even, hence $\#_b(w)$ is even; in both cases, $w \notin O_{n+1}$. If $w = bz$ with $z \in \Sigma^* \setminus E_n$, then either $|z| \geq n$, hence $|w| \geq n + 1$, or $|z| < n$ and $\#_b(z)$ is odd, hence $\#_b(w)$ is even; again $w \notin O_{n+1}$. Thus $X_2^{(n+1)} \subseteq \Sigma^* \setminus O_{n+1}$.

Inclusion $\Sigma^* \setminus O_{n+1} \subseteq X_2^{(n+1)}$. Let $w \notin O_{n+1}$. If $w = \varepsilon$, then clearly $w \in X_2^{(n+1)}$. Assume $w \neq \varepsilon$. Again, w begins either with a or with b. If $w = az$, then either $|w| \geq n + 1$, hence $|z| \geq n$, so $z \notin O_n$, or $|w| < n + 1$ and $w \notin O_{n+1}$, so w has even b-parity; in the latter case z also has even b-parity and $|z| < n$, hence again $z \notin O_n$. Thus $w \in a(\Sigma^* \setminus O_n) \subseteq X_2^{(n+1)}$.

 doi:10.20944/preprints202605.1259.v1

If $w = bz$, then either $|w| \geq n+1$, hence $|z| \geq n$, so $z \notin E_n$, or $|w| < n+1$ and $w \notin O_{n+1}$, so w has even b-parity; in the latter case z has odd b-parity and $|z| < n$, hence again $z \notin E_n$. Thus $w \in b(\Sigma^* \setminus E_n) \subseteq X_2^{(n+1)}$.

Therefore $\Sigma^* \setminus O_{n+1} \subseteq X_2^{(n+1)}$. Combining the two inclusions, we conclude that $X_2^{(n+1)} = \Sigma^* \setminus O_{n+1}$, hence $(X_2^{(n+1)})^c = O_{n+1}$. This completes the induction.

Step 3: passage to the limit.

We have proved that, for every $n \geq 1$, $X_1^{(n)} = \{\, w \in \Sigma^* : |w| < n \text{ and } \#_b(w) \text{ is even} \,\}$. Since the sequence $(X^{(n)})$ is increasing and converges to the unique fixed pair by Theorem 2, its first coordinate converges to $L^* = \bigcup_{n\geq 1} X_1^{(n)}$. Hence $L^* = \{\, w \in \{a,b\}^* : \#_b(w) \text{ is even} \,\}$. Because the fixed pair is diagonal, the unique fixed pair of the system is (L^*, L^*).

Conclusion. Starting from the canonical lower pair $X^{(0)} = (\varnothing, \Sigma^*)$, the first coordinate $X_1^{(n)}$ generates exactly the words of length $< n$ having even b-parity. Thus the canonical Picard iteration constructs the fixed point level by level with respect to word length, and its limit is precisely the language $L^* = \{\, w \in \{a,b\}^* : \#_b(w) \text{ is even} \,\}$.

4. Application: A Semantics of Trace-Based Policies for Tool-Using AI Agents

The positive–negative guarded language framework developed in the previous sections has a natural interpretation in the formal specification of policies for tool-using AI agents. In such systems, an agent does not merely generate text. It may also perform externally observable actions, such as reading files, querying databases, sending emails, filling web forms, executing code, exporting reports, or invoking external services. After a suitable abstraction, an execution of the agent may be represented as a finite word over an alphabet of observable tool-events.

Thus, let Σ be a finite alphabet of abstract tool-events. For example, one may consider events such as

`read_sensitive_file`, `summarize`, `send_external_email`,
`download_untrusted_code`, `verify_code`, `sandbox`, `privileged_run`,
`query_customer_database`, `anonymize`, `call_external_api`,
`human_approve`, `redact_sensitive_data`, `audit_log`.

A finite execution trace of the agent is then a word $w \in \Sigma^*$. For instance,

`read_sensitive_file summarize send_external_email`

and

`download_untrusted_code sandbox privileged_run`

are examples of finite tool-call traces over such an alphabet.

The purpose of the present interpretation is not to model the internal reasoning process of the language model, nor its statistical behavior. The framework does not aim to improve the agent's planning ability, training procedure, reasoning capacity, or quality of generated responses. Its role is narrower and more formal: it provides a fixed point semantics for policies imposed on the externally observable sequence of tool-actions proposed or executed by the agent.

In this setting, formal languages are interpreted as sets of tool-call traces. A policy over agent behavior may therefore be represented by policy-defined trace classes. In particular, the two components of the fixed pair from the previous sections may be interpreted as follows:

$$A^* = \text{the language of traces marked as admissible by the policy,}$$

and

$$B^* = \text{the language of traces marked as risky, blocked, or review-relevant by the policy.}$$

Thus, the fixed pair $\mathbf{L}^* = (A^*, B^*)$ represents a stable fixed-point semantics for two policy-defined trace classes.

It is important to emphasize that the general fixed point theorem does not, by itself, imply that these two trace classes are disjoint or exhaustive; in particular, it does not automatically imply $A^* \cap B^* = \emptyset$ or $A^* \cup B^* = \Sigma^*$. Therefore, the fixed pair should not be interpreted as a complete and disjoint safe-versus-risky classification unless additional consistency or exhaustiveness assumptions are imposed on the policy rules. The framework provides a stable fixed-point semantics for policy-defined trace classes; a strict disjoint or total classification would require extra hypotheses.

4.1. A Guarded Trace-Policy Operator

Let $A \subseteq \Sigma^*$ and $B \subseteq \Sigma^*$ denote current approximations of the traces marked by the policy as admissible and as risky, blocked, or review-relevant, respectively. We define the positive–negative guarded policy operator $\mathbf{T}(A, B) = \big(T_{\text{adm}}(A, B), T_{\text{risk}}(A, B)\big)$, where T_{adm} updates the policy-admissible traces and T_{risk} updates the traces marked as risky, blocked, or review-relevant by the policy.

A general instance of such an operator has the form

$$T_{\text{adm}}(A, B) = S_{\text{adm}} \cup \bigcup_{(u,v)\in\mathcal{G}^+_{\text{adm}}} uAv \cup \bigcup_{(u,v)\in\mathcal{G}^-_{\text{adm}}} uB^c v,$$
$$T_{\text{risk}}(A, B) = S_{\text{risk}} \cup \bigcup_{(u,v)\in\mathcal{G}^-_{\text{risk}}} uA^c v \cup \bigcup_{(u,v)\in\mathcal{G}^+_{\text{risk}}} uBv.$$

Here S_{adm} and S_{risk} are seed languages of traces immediately marked by the policy as admissible and as risky, blocked, or review-relevant, respectively. The finite families $\mathcal{G}^+_{\text{adm}}$, $\mathcal{G}^-_{\text{adm}}$, $\mathcal{G}^-_{\text{risk}}$, and $\mathcal{G}^+_{\text{risk}}$ consist of guarded contexts. Positive terms of the form uAv and uBv propagate policy-admissibility or policy-risk through observable tool-event contexts, while negative terms of the form uB^cv and uA^cv express policy conditions depending on the absence of risk or admissibility in the current approximation.

For example, a term of the form

$$\texttt{read\_sensitive\_file}\, A^c\, \texttt{send\_external\_email}$$

belongs naturally to $T_{\text{risk}}(A, B)$. It expresses the following policy principle: a trace in which a sensitive file is read and information is sent externally is marked as risky under the policy when the intermediate continuation is not known to be admissible. More explicitly, if

$$w = \texttt{read\_sensitive\_file}\, z\, \texttt{send\_external\_email},$$

then this rule marks w as risky whenever $z \notin A$.

Similarly, a term of the form

$$\texttt{human\_approve}\, B^c$$

may be placed in $T_{\text{adm}}(A, B)$. It expresses that, after human approval, a continuation may be admitted provided that it is not marked as risky by the policy. In this way, negative guarded rules allow one to express policies of the form "an action context is admissible only when the continuation is not risky under the specified policy".

The minimal guard length of the policy is

$$m = \min\{|u| + |v| : (u, v) \in \mathcal{G}^+_{\text{adm}} \cup \mathcal{G}^-_{\text{adm}} \cup \mathcal{G}^-_{\text{risk}} \cup \mathcal{G}^+_{\text{risk}}\}.$$

The policy is guarded whenever $m \geq 1$. This means that every recursive occurrence of A, B, A^c, or B^c is separated from the whole trace by at least one observable tool-event.

 doi:10.20944/preprints202605.1259.v1

4.2. Connection with the Fixed Point Theorem

The operator $\mathbf{T}$ above is a direct instance of the positive–negative guarded system studied in the previous sections. Indeed, T_{adm} plays the role of T_1, and T_{risk} plays the role of T_2. The mixed order $(A_1, B_1) \preceq (A_2, B_2)$ means that $A_1 \subseteq A_2$ and $B_2 \subseteq B_1$. It has the following policy interpretation: in the first coordinate, more elements mean more traces certified as admissible by the policy, while in the second coordinate, the reverse inclusion reflects the fact that fewer traces remaining in the risk component represent a stronger approximation from above.

The complement operation is essential in such policies. Since complement is an isometry for the length-based ultrametric,$d(A^c, B^c) = d(A, B)$, negative policy conditions do not destroy the ultrametric estimates. Thus, if two approximations of admissible traces agree up to a given trace length, then their complements also agree up to the same trace length.

Moreover, for every guarded policy context $u(\cdot)v$, the signed wrapping estimate gives

$$d(uL^\sigma v, uM^\sigma v) \le 2^{-m} d(L, M), \qquad \sigma \in \{+, -\}.$$

In the present interpretation, this means that placing a policy condition inside a nonempty context of tool-actions delays the first possible disagreement by at least the length of the guard. Therefore guarded trace rules are contractive in the length-based ultrametric.

Consequently, the general fixed point theorem applies. The canonical lower and upper initial pairs are $\mathbf{X}^{(0)} = (\emptyset, \Sigma^*)$ and $\mathbf{Y}^{(0)} = (\Sigma^*, \emptyset)$. The lower initial pair represents a pessimistic policy approximation: initially, no trace is certified as admissible, while every trace is treated as potentially risky. The upper initial pair represents an optimistic approximation: initially, every trace is potentially admissible, while no trace has yet been marked as risky.

The canonical Picard iterations are $\mathbf{X}^{(n+1)} = \mathbf{T}(\mathbf{X}^{(n)})$ and $\mathbf{Y}^{(n+1)} = \mathbf{T}(\mathbf{Y}^{(n)})$. By Theorem 2, these two iterations converge to the same fixed pair $\mathbf{L}^* = (A^*, B^*)$. Thus the formal policy has a unique stable fixed-point semantics. In particular, the resulting policy-defined trace classes do not depend on an arbitrary order of rule application, nor on whether the approximation starts from a pessimistic or an optimistic initial state.

This is the main semantic contribution of the framework in the present application. It provides sufficient mathematical conditions under which a policy with mutually dependent positive and negative trace rules has a unique stable fixed-point interpretation.

4.3. An Expressive Trace-Policy Example

We now give a more detailed trace-policy example whose purpose is to show why the fixed point framework is useful beyond purely local rule checking. The example involves three interacting sources of policy risk: possible data exfiltration, unsafe code execution, and external transmission of database information. The important point is that the policy status of a trace is not determined only by its last action or by a single forbidden pattern. Rather, admissibility and risk depend on mutually recursive positive and negative conditions.

Let

$$\Sigma = \{r, z, e, d, v, s, p, q, a, x, h, t, g\},$$

where

$$\begin{aligned}
&r = \texttt{read_sensitive_file}, \quad z = \texttt{summarize}, \quad e = \texttt{send_external_email}, \\
&d = \texttt{download_untrusted_code}, \quad v = \texttt{verify_code}, \quad s = \texttt{sandbox}, \quad p = \texttt{privileged_run}, \\
&q = \texttt{query_customer_database}, \quad a = \texttt{anonymize}, \quad x = \texttt{call_external_api}, \\
&h = \texttt{human_approve}, \quad t = \texttt{redact_sensitive_data}, \quad g = \texttt{audit_log}.
\end{aligned}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 May 2026 doi:10.20944/preprints202605.1259.v1

Thus, for example, the word rze represents the trace

`read_sensitive_file summarize send_external_email`,

while $dvsp$ represents the trace

`download_untrusted_code verify_code sandbox privileged_run`.

Consider the following guarded trace-policy operator:

$$\mathbf{T}(A,B) = \big(T_{\text{adm}}(A,B), T_{\text{risk}}(A,B)\big),$$

where

$$\begin{aligned} T_{\text{adm}}(A,B) &= S_{\text{adm}} \cup hB^c \cup aB^c \cup tB^c \cup gA \cup vA \cup sA \cup tzA \\ &\quad \cup rtB^c e \cup qaB^c x \cup dvsAp, \\ T_{\text{risk}}(A,B) &= S_{\text{risk}} \cup rA^c e \cup dA^c p \cup qA^c x \\ &\quad \cup rBe \cup dBp \cup qBx \cup zB \cup gB. \end{aligned}$$

Here S_{adm} is a seed language of traces that are immediately admissible under the policy, and S_{risk} is a seed language of traces that are immediately marked as risky, blocked, or review-relevant under the policy. For instance, one may assume that $\varepsilon \in S_{\text{adm}}$, so that the empty continuation is admissible, and that $re,\ rze,\ dp,\ qx \in S_{\text{risk}}$, so that some basic unsafe patterns are marked as risky by the policy from the beginning.

The terms in T_{adm} have the following interpretation. The terms hB^c, aB^c, and tB^c express that human approval, anonymization, and redaction may make a continuation admissible when that continuation is not marked as risky by the policy. The terms gA, vA, and sA express positive propagation of admissibility through audit logging, code verification, and sandboxing. The term tzA expresses that summarization is admissible when it is preceded by redaction and followed by an admissible continuation.

The longer guarded terms express genuinely nonlocal policy principles. The term $rtB^c e$ says that a trace in which a sensitive file is read, then redacted, and then sent externally may be admissible provided that the intermediate continuation is not marked as risky by the policy. The term $qaB^c x$ says that querying a customer database and then calling an external API may be admissible if anonymization has occurred and the intermediate continuation is not marked as risky. Finally, the term $dvsAp$ says that privileged execution after downloading code may be admissible only when verification and sandboxing have occurred and the remaining continuation is admissible.

The terms in T_{risk} express the dual risk-propagation principles. The term $rA^c e$ says that reading a sensitive file and later sending information externally is marked as risky when the intermediate continuation is not known to be admissible. The term $dA^c p$ says that running downloaded code with privileges is marked as risky when the intermediate continuation is not known to be admissible. The term $qA^c x$ says that querying a customer database and later calling an external API is marked as risky when the intermediate continuation is not admissible. The terms rBe, dBp, and qBx propagate already detected policy risk through sensitive contexts. The terms zB and gB express that summarization and audit logging do not remove risk by themselves.

This example shows why the positive–negative structure is essential. Rules such as hB^c, aB^c, $rtB^c e$, and $qaB^c x$ are negative with respect to the risk language B, whereas rules such as $rA^c e$, $dA^c p$, and $qA^c x$ are negative with respect to the admissibility language A. Hence the policy is not monotone with respect to the ordinary product order. The mixed order used in this paper is precisely the order that supports monotone lower and upper approximations and gives an order-theoretic interpretation of the positive–negative dependencies.

 doi:10.20944/preprints202605.1259.v1

The example is also genuinely recursive. Whether a trace is admissible may depend on whether another trace is not risky under the policy, while whether a trace is risky may depend on whether another trace is not admissible. Thus the classes "admissible" and "risky under the specified policy" are not defined independently. They are defined simultaneously as a fixed pair of the operator $\mathbf{T}$. The uniqueness of this fixed pair is ultimately guaranteed by the guarded metric contraction, while the mixed order provides the monotone two-sided approximation scheme and the policy interpretation of the positive–negative dependencies.

All recursive occurrences in the displayed operator are guarded. The global minimal guard length is $m = 1$, because terms such as hB^c, aB^c, vA, and sA have guard length one. Some of the more important nonlocal rules have larger local guard lengths: $|rt| + |e| = 3$, $|qa| + |x| = 3$, and $|dvs| + |p| = 4$. Thus the general theorem gives the conservative global contraction factor 2^{-1}, while the longer policy contexts delay possible disagreements even more strongly.

By Theorem 2, the operator $\mathbf{T}$ has a unique fixed pair (A^*, B^*). This pair is the unique stable fixed-point semantics of the formal policy. In particular, it does not depend on whether one starts from the pessimistic approximation $(\varnothing, \Sigma^*)$ or from the optimistic approximation $(\Sigma^*, \varnothing)$. Both canonical Picard iterations converge to the same pair (A^*, B^*).

4.4. Why the Example Is Not Merely a Finite List of Forbidden Traces

The preceding policy is not just a finite blacklist of undesirable traces. A finite blacklist could mark traces such as re, rze, dp, and qx as risky under the policy. However, it would not by itself explain how risk propagates through larger contexts, nor how neutralizing operations such as redaction, anonymization, sandboxing, verification, and human approval interact with recursive policy conditions.

For example, the two traces

$$rze \qquad \text{and} \qquad rtze$$

should not necessarily have the same policy status. The first trace represents reading a sensitive file, summarizing it, and sending the result externally. The second trace contains an additional redaction step before summarization and external transmission. In the displayed operator, this difference is expressed through the admissibility rule tzA, the negative admissibility rule rtB^ce, and the risk rule rA^ce. The status of $rtze$ depends on whether the intermediate continuation tz is admissible and not risky under the policy, while the status of rze depends on whether z is admissible. This is a nonlocal trace property.

Similarly, the traces

$$dp \qquad \text{and} \qquad dvsp$$

are treated differently by the policy. The trace dp represents running downloaded untrusted code with privileges without verification or sandboxing, and may be placed directly in S_{risk}. The trace $dvsp$, on the other hand, contains verification and sandboxing before privileged execution. Its admissibility is represented by the guarded rule $dvsAp$. Thus the policy does not simply prohibit all occurrences of privileged execution after downloading code; it distinguishes whether a suitable protected context has occurred.

A third example is the difference between

$$qx \qquad \text{and} \qquad qax.$$

The first trace represents querying a customer database and then calling an external API. The second trace inserts anonymization before the external API call. The rule qaB^cx expresses that the second trace may be admissible when the intermediate continuation is not risky under the policy, while the seed language S_{risk} may mark qx as risky from the beginning.

These examples illustrate the role of guarded contexts. The theory does not only mark isolated forbidden words. It gives a semantics for recursive policies where the policy status of a trace depends on the policy status of its subtraces.

4.5. Two-Sided Policy Approximation and Uncertainty Regions

The canonical lower and upper iterations also have a useful policy interpretation. Let $\mathbf{X}^{(n)} = (X^{(n)}_{\text{adm}}, X^{(n)}_{\text{risk}})$ be the iteration starting from $\mathbf{X}^{(0)} = (\emptyset, \Sigma^*)$, and let $\mathbf{Y}^{(n)} = (Y^{(n)}_{\text{adm}}, Y^{(n)}_{\text{risk}})$ be the iteration starting from $\mathbf{Y}^{(0)} = (\Sigma^*, \emptyset)$. Then Theorem 2 gives $\mathbf{X}^{(n)} \preceq (A^*, B^*) \preceq \mathbf{Y}^{(n)}$ for all $n \geq 0$. Equivalently, $X^{(n)}_{\text{adm}} \subseteq A^* \subseteq Y^{(n)}_{\text{adm}}$ and $Y^{(n)}_{\text{risk}} \subseteq B^* \subseteq X^{(n)}_{\text{risk}}$. Thus $X^{(n)}_{\text{adm}}$ consists of traces already certified as admissible with respect to the formal policy semantics, while $Y^{(n)}_{\text{adm}}$ consists of traces still possibly admissible. Dually, $Y^{(n)}_{\text{risk}}$ consists of traces already marked as risky by the formal policy semantics, while $X^{(n)}_{\text{risk}}$ consists of traces still possibly risky.

This gives natural uncertainty regions: $\mathcal{U}^{(n)}_{\text{adm}} = Y^{(n)}_{\text{adm}} \setminus X^{(n)}_{\text{adm}}$ and $\mathcal{U}^{(n)}_{\text{risk}} = X^{(n)}_{\text{risk}} \setminus Y^{(n)}_{\text{risk}}$. The first set contains traces whose admissibility status has not yet stabilized at iteration n, and the second contains traces whose risk status has not yet stabilized.

This is where the finite-depth estimate becomes operationally meaningful. If $k \geq \lceil N/m \rceil$, then Corollary 7 implies $\mathcal{U}^{(k)}_{\text{adm}} \cap \Sigma^{<N} = \emptyset$ and $\mathcal{U}^{(k)}_{\text{risk}} \cap \Sigma^{<N} = \emptyset$. In other words, after k iterations there is no remaining uncertainty, with respect to the formal fixed-point policy semantics, for traces of length strictly less than N. All such traces have stabilized with respect to the unique fixed point semantics.

In the present example $m = 1$, so $k = N$ iterations are sufficient to certify all traces of length less than N. Although this bound is conservative, it is explicit and independent of the total number of possible traces beyond depth N.

4.6. Finite-Depth Certification

A central advantage of the length-based ultrametric is that convergence has a finite-depth interpretation. In realistic tool-using agent systems, it is often not necessary to evaluate all possible traces of arbitrary length. A more practical goal is to evaluate the policy status of all traces up to a fixed length N, for example all traces involving fewer than N tool calls.

By Corollary 7, if $n \geq \lceil N/m \rceil$, then the n-th canonical approximation agrees with the fixed point semantics on all traces of length strictly less than N. In other words, for every $w \in \Sigma^*$ with $|w| < N$, we have $\mathbf{1}_{A^{(n)}}(w) = \mathbf{1}_{A^*}(w)$ and $\mathbf{1}_{B^{(n)}}(w) = \mathbf{1}_{B^*}(w)$, where $(A^{(n)}, B^{(n)})$ denotes the corresponding canonical approximation.

Thus, after sufficiently many iterations, the policy-defined membership status of every trace of length less than N has stabilized. In the special case $m = 1$, it is sufficient to perform N iterations. More generally, after n iterations all traces of length less than mn have the same policy-defined membership status as in the final fixed point semantics.

This gives a concrete certification statement: all tool-call traces of length $< N$ have the same policy status as in the fixed point semantics. The result is not merely heuristic. It follows directly from the contraction estimate $d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$. Since $d_\infty \leq 2^{-N}$ implies agreement on all words of length less than N, the quantitative convergence estimate becomes a finite-depth certificate for the formal policy semantics.

4.7. Finite-Depth Policy Evaluation in the Example

For a fixed verification depth N, the set $\Sigma^{<N} = \{w \in \Sigma^* : |w| < N\}$ is finite. The canonical iteration therefore gives a finite procedure for evaluating the policy status of all traces up to depth N. After $k = \lceil N/m \rceil$ iterations, membership in the approximants agrees with membership in the fixed pair (A^*, B^*) for every word of length less than N.

A conservative policy-status interpretation can be defined as follows: $w \in A^* \setminus B^*$ implies policy-admissible, $w \in B^*$ implies blocked or review-required under the policy, and $w \notin A^* \cup B^*$ implies unclassified or policy-incomplete. This convention treats policy risk conservatively: if a trace belongs

 doi:10.20944/preprints202605.1259.v1

to B^*, then it is blocked or sent to review, even if it also belongs to A^*. If one wants to exclude overlaps $A^* \cap B^*$, additional consistency assumptions on the seed languages and guarded rules must be imposed. Such assumptions are separate from the existence and uniqueness theorem.

For the policy above, a finite-depth trace evaluator may produce policy-status information of the following kind:

Table 1. Illustrative finite-depth policy status of traces for the guarded agent policy.

Trace	Policy status
re	blocked or review-required under the policy
rze	blocked or review-required under the policy
$rtze$	policy-admissible or review-dependent
dp	blocked or review-required under the policy
$dvsp$	policy-admissible or review-dependent
qx	blocked or review-required under the policy
qax	policy-admissible or review-dependent
he	admissible if the continuation is not risky under the policy

The entries in the table are illustrative: their precise policy status depends on the chosen seed languages and on the full family of guarded rules. The point is not that the displayed table is itself the policy, but that the fixed point semantics gives a mathematically controlled way to evaluate policy status up to a prescribed trace depth.

4.8. Algorithmic Interpretation

The theory suggests the following finite-depth procedure for trace-based policy evaluation.

(1) Choose a finite abstraction alphabet Σ of observable tool-events.
(2) Map concrete tool calls to abstract events in Σ.
(3) Specify seed languages S_{adm} and S_{risk}.
(4) Specify finite guarded rule families $\mathcal{G}^+_{\text{adm}}$, $\mathcal{G}^-_{\text{adm}}$, $\mathcal{G}^-_{\text{risk}}$, and $\mathcal{G}^+_{\text{risk}}$.
(5) Compute the minimal guard length $m = \min\{|u| + |v| : (u, v) \in \mathcal{G}\}$.
(6) Choose a verification depth N.
(7) Compute $k = \lceil N/m \rceil$.
(8) Run the canonical Picard iteration for k steps.
(9) Evaluate the policy status of all traces $w \in \Sigma^{<N}$ using the obtained approximation.

For each fixed depth N, the set $\Sigma^{<N} = \{w \in \Sigma^* : |w| < N\}$ is finite. Hence the computation up to depth N can be implemented as a finite Boolean table. Complements are evaluated only on the finite prefix-depth universe relevant to the chosen N. Therefore, even if the final languages A^* and B^* are infinite, the certified finite-depth approximation is computationally finite.

4.9. What the Fixed Point Theorem Contributes

The mathematical contribution of the theorem in this application can be summarized in four points.

First, the theorem gives existence of a stable fixed-point policy semantics. Since the operator contains negative occurrences through A^c and B^c, the policy is not a purely positive inductive definition. The theorem shows that, under guardedness, the simultaneous definition of admissible and risky trace classes is still well posed.

Second, the theorem gives uniqueness. Without uniqueness, different ways of resolving the positive and negative dependencies might lead to different policy interpretations. The guarded metric contraction rules this out: the stable fixed pair (A^*, B^*) is unique.

Third, the theorem gives a canonical approximation procedure. The lower and upper Picard iterations provide two-sided approximations of the same semantic object. This is useful in policy analysis because it separates traces already certified with respect to the formal policy semantics from traces that are still only possibly admissible or possibly risky under the policy.

Fourth, the theorem gives a finite-depth certificate. For every prescribed trace length N, after $\lceil N/m \rceil$ iterations, all traces of length less than N have the same policy status with respect to the formal fixed point semantics. Thus the method does not merely iterate rules heuristically. It gives an explicit mathematical guarantee of stabilization of the policy semantics up to the chosen depth.

This is the main practical value of the framework. It is not intended to replace access-control systems, human approval mechanisms, content classifiers, data-loss-prevention tools, sandboxing, or code-analysis systems. Rather, it provides a semantic layer in which complex trace-based policies with mutually recursive positive and negative conditions can be specified, approximated, and certified up to finite depth.

4.10. Scope and Limitations

The proposed application should be understood as a semantic layer for trace-based policies, not as a complete technology for controlling AI agents. Several limitations should be made explicit.

First, concrete tool calls must be abstracted to a finite alphabet of events. For example, a concrete action such as sending an email to an external recipient with sensitive content must be represented by an abstract symbol such as

`send_external_sensitive_email`.

The present theory does not perform this abstraction by itself.

Second, the framework does not analyze content. It does not decide whether a file is sensitive, whether text contains personal data, whether code is malicious, or whether a website is untrusted. Such tasks require additional mechanisms, such as content classifiers, data-loss-prevention systems, information-flow tracking, code analysis, sandboxing, or human review.

Third, the framework is not intended for simple local policies. If the only rule is, for example, "sending an email requires human approval", then a direct approval mechanism, a finite automaton, or a workflow graph may be more suitable. The advantage of the present theory appears in more complex situations where the status of a trace depends on recursive and negative conditions involving other trace classes.

Thus, the contribution of the framework in this application is semantic rather than operational. It provides a mathematically well-defined fixed point semantics for nonlocal, mutually recursive policy-defined trace classes. The finite-depth certification theorem then gives a precise guarantee that, up to any prescribed trace length, the computed approximation agrees with the unique stable fixed-point semantics of the policy.

In this sense, positive–negative guarded language systems provide a bridge between ultrametric fixed point theory and the formal specification of trace-based policies for tool-using AI agents.

5. Conclusion and Future Work

In this paper, we developed a fixed point framework for positive–negative guarded systems of formal language equations in a length-based ultrametric space. In contrast with purely positive recursive language operators, the systems considered here allow both ordinary recursive dependencies and dependencies through language complements. This leads naturally to a mixed order structure on pairs of languages, where one coordinate is ordered by inclusion and the other by reverse inclusion.

The main point of the approach is that guardedness still produces contraction-type behavior, even in the presence of negative occurrences. We showed that the complement operation is an isometry with respect to the length-based ultrametric and proved a signed wrapping estimate for both positive and negative guarded language terms. Combining these estimates with the finite-union stability of the ultrametric, we obtained an ordered contraction principle for comparable pairs.

Using this guarded contraction structure together with the mixed order, we proved that the canonical lower and upper Picard iterations converge to the same limit. This common limit is the

doi:10.20944/preprints202605.1259.v1

unique fixed pair of the corresponding positive–negative guarded language system. Thus, the length-based ultrametric and guardedness provide the metric mechanism behind uniqueness, while the mixed order supports the monotone two-sided approximation scheme for systems involving complements.

A central feature of the result is its quantitative character. The convergence rate is explicitly controlled by the minimal guard length m. In particular, the estimate

$$d_\infty(\mathbf{L}^{(n)}, \mathbf{L}^*) \leq 2^{-mn} d_\infty(\mathbf{L}^{(0)}, \mathbf{L}^*)$$

shows that guardedness has a direct finite-depth interpretation. After sufficiently many iterations, the approximating languages agree with the fixed pair on all words below a prescribed length. For the canonical lower or upper initial pair, $n \geq \lceil N/m \rceil$ iterations are sufficient to guarantee agreement with the fixed-point semantics on all words of length less than N.

We also studied symmetry properties of the system. When the two components satisfy a structural symmetry condition, the system commutes with the coordinate-swap map. Under uniqueness, this forces the fixed pair to be diagonal and reduces the two-component system to a single language equation. This provides a useful criterion for identifying symmetric equilibria in positive–negative guarded systems.

The application to trace-based policies for tool-using AI agents illustrates how the abstract theory may be interpreted in a semantic setting. There, words represent finite traces of observable tool-events, while the two components of the fixed pair represent policy-defined admissible and risky trace classes. Positive rules propagate admissibility or risk, whereas negative rules model conditions depending on the absence of membership in the opposite trace class. The finite-depth certification theorem then gives a mathematical guarantee that the computed approximation agrees with the stable fixed-point policy semantics for all traces up to a prescribed length.

It is important to stress that this application is semantic rather than operational. The framework does not prove real-world safety of a tool-using AI agent, nor does it replace access-control mechanisms, human approval, sandboxing, content classifiers, code analysis, or data-loss-prevention tools. Instead, it provides a mathematically well-defined semantics for recursive trace-based policies once the relevant tool-events, seed trace classes, and guarded rules have been specified.

The present work suggests several directions for future research. A natural extension is to consider systems with more than two components, corresponding to several policy states, user roles, tool contexts, or semantic categories. Another direction is to study weighted product ultrametrics, where different components may have different importance or different stabilization speeds. This could lead to sharper convergence estimates for asymmetric systems.

It would also be useful to investigate automata-theoretic versions of the present framework. Since formal languages are often represented by finite automata, one may ask when the fixed pair of a positive–negative guarded system is regular, context-free, or effectively approximable by finite-state models. Such questions could connect the present ultrametric fixed point approach with algorithmic verification, model checking, and symbolic computation.

Another promising direction is the development of finite-depth algorithms based directly on the theoretical estimates proved here. For each fixed depth N, the set $\Sigma^{<N}$ is finite, so the canonical iterations may be implemented as finite Boolean tables. This suggests the possibility of prototype tools for guarded trace-policy analysis, finite-depth policy evaluation, and bounded validation of recursive specifications.

Finally, the ordered nature of the present construction suggests possible links with broader fixed point theories in partially ordered metric and ultrametric spaces, including weak monotone mappings, coupled fixed points, and multi-variable contractive conditions. Extending the present method to more general contractive-type assumptions may allow the treatment of recursive language systems with richer dependencies and weaker forms of guardedness.

We hope that the results of this paper provide a useful step toward a systematic interaction between fixed point theory, ultrametric methods, formal language theory, and finite-depth semantics for recursive trace-based policies.

Author Contributions: The mentioned authors participated equally to the study and are arranged in alphabetical order as follows: conceptualization, L.A., H.H., A.I., and B.Z.; methodology, L.A., H.H., A.I., and B.Z.; investigation, L.A., H.H., A.I., and B.Z.; writing—original draft preparation, L.A., H.H., A.I., and B.Z.; writing—review and editing, L.A., H.H., A.I., and B.Z. All authors have read and agreed to the published version of the manuscript.

Acknowledgments: The authors would like to express their gratitude for the support provided by the European Union-NextGenerationEU, through the National Recovery and Resilience Plan of the Republic of Bulgaria, project DUECOS BG-RRP-2.004-0001-C01.

Funding: This research received no external funding.

Data Availability Statement: Data sharing is not applicable to this article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Banach, S. Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales. *Fundam. Math.* **1922**, *3*, 133–181.
2. America, P.; Rutten, J.J.M.M. Solving reflexive domain equations in a category of complete metric spaces. *J. Comput. Syst. Sci.* **1989**, *39*, 343–375. https://doi.org/10.1016/0022-0000(89)90027-5
3. Vianu, V. The Bodnarchuk Metric Space of Languages and the Topology of the Learning Space. In *Mathematical Foundations of Computer Science 1977*; Gruska, J., Ed.; Lecture Notes in Computer Science, Vol. 53; Springer: Berlin/Heidelberg, Germany, 1977; pp. 537–542. https://doi.org/10.1007/3-540-08353-7_177
4. Fulop, S.A.; Kephart, D. Topology of Language Classes. In *Proceedings of the 14th Meeting on the Mathematics of Language*; Morrill, G.; Muskens, R.; Osswald, R.; Richter, F., Eds.; Association for Computational Linguistics: Chicago, IL, USA, 2015; pp. 26–38. https://aclanthology.org/W15-2303/
5. Salomaa, A. *Formal Languages*; Academic Press: New York, NY, USA, 1973; ISBN 978-0-12-615750-5.
6. Eilenberg, S. *Automata, Languages, and Machines, Vol. A*; Academic Press: New York, NY, USA, 1974; ISBN 978-0-12-234001-7.
7. Schikhof, W.H. *Ultrametric Calculus: An Introduction to p-Adic Analysis*. Cambridge University Press: Cambridge, UK, 1984. https://doi.org/10.1017/CBO9780511623844
8. Bhaskar, T.G.; Lakshmikantham, V. Fixed point theorems in partially ordered metric spaces and applications. *Nonlinear Anal. Theory Methods Appl.* **2006**, *65*, 1379–1393. https://doi.org/10.1016/j.na.2005.10.017
9. Ran, A.C.M.; Reurings, M.C.B. A fixed point theorem in partially ordered sets and some applications to matrix equations. *Proc. Am. Math. Soc.* **2004**, *132*, 1435–1443. https://doi.org/10.1090/S0002-9939-03-07220-4
10. Guo, D.; Lakshmikantham, V. Coupled fixed points of nonlinear operators with applications. *Nonlinear Anal.* **1987**, *11*, 623–632. https://doi.org/10.1016/0362-546X(87)90077-0
11. Zlatanov, B. Coupled best proximity points for cyclic contractive maps and their applications. *Fixed Point Theory* **2021**, *22*, 431–452. https://doi.org/10.24193/fpt-ro.2021.2.32
12. Kabaivanov, S.; Zlatanov, B. A variational principle, coupled fixed points and market equilibrium. *Nonlinear Anal. Model. Control* **2021**, *26*, 169–185. https://doi.org/10.15388/namc.2021.26.22050
13. Ali, A.; Dinkova, C.; Kulina, H.; Ilchev, A.; Zlatanov, B. Positive definite solution of system of matrix equations with X^k and Y^l via coupled fixed point theorem in partially ordered spaces. *Axioms* **2025**, *14*, 123. https://doi.org/10.3390/axioms14020123