Math. Control Signals Syst. (2017) 29:21
https://doi.org/10.1007/s00498-017-0207-8

ORIGINAL ARTICLE

Dynamical systems on weighted lattices: general theory

Petros Maragos[1]

Received: 31 July 2016 / Accepted: 22 November 2017 / Published online: 16 December 2017

Abstract In this work, a theory is developed for unifying large classes of nonlinear discrete-time dynamical systems obeying a superposition of a weighted maximum or minimum type. The state vectors and input–output signals evolve on nonlinear spaces which we call complete weighted lattices and include as special cases the nonlinear vector spaces of minimax algebra. Their algebraic structure has a polygonal geometry. Some of the special cases unified include max-plus, max-product, and probabilistic dynamical systems. We study problems of representation in state and input–output spaces using lattice monotone operators, state and output responses using nonlinear convolutions, solving nonlinear matrix equations using lattice adjunctions, stability, and controllability. We outline applications in state-space modeling of nonlinear filtering; dynamic programming (Viterbi algorithm) and shortest paths (distance maps); fuzzy Markov chains; and tracking audiovisual salient events in multimodal information streams using generalized hidden Markov models with control inputs.

Keywords Nonlinear dynamical systems · Lattice theory · Minimax algebra · Control · Signal processing

This research was supported by the project “COGNIMUSE” under the “ARISTEIA” Action of the Operational Program Education and Lifelong Learning and was co-funded by the European Social Fund and Greek National Resources. It was also partially supported by the European Union under the projects MOBOT with Grant FP7-600796 and BabyRobot with Grant H2020-687831.

✉ Petros Maragos
maragos@cs.ntua.gr

[1] School of Electrical and Computer Engineering, National Technical University of Athens, Zografou Campus, 15773 Athens, Greece

1 Introduction

Linear dynamical systems [12,14,36] can be described in discrete-time by the state-space equations

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{A}\boldsymbol{x}(t-1) + \boldsymbol{B}\boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}\boldsymbol{x}(t) + \boldsymbol{D}\boldsymbol{u}(t) \end{aligned} \tag{1}$$

where $t \in \mathbb{Z}$ shall denote a discrete-time index, $\boldsymbol{x}(t)$ is an evolving state vector, $\boldsymbol{u}(t)$ is the input signal (scalar or vector), and $\boldsymbol{y}(t)$ is an output signal (scalar or vector). $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$ are appropriately sized matrices, and all the matrix–vector products are defined in the standard linear way. Linear systems have proven useful for a plethora of problems in communications, control, and signal processing. The strongest motivation for using linear systems as models has been the great familiarity of all sciences with linear mathematics, e.g., linear algebra, linear vector spaces, and linear differential equations, as well as the availability of computational tools and algorithms to solve problems with linear systems.

However, in the 1980s and 1990s, several broad classes of *nonlinear* systems were developed whose state-space dynamics can be described by equations whose structure resembles (1) but has nonlinear operations. These were motivated by a broad spectrum of applications, such as scheduling and synchronization, operations research, dynamic programming, shortest paths on graphs, image processing, and non-Gaussian estimation, for which nonlinear systems were more appropriate. These nonlinearities involve two major elements: (1) a nonlinear superposition of vectors/matrices via pointwise maximum ($\vee$) or minimum ($\wedge$) which plays the role of a generalized 'addition,' and (2) a binary operation $\star$ among scalars that plays the role of a generalized 'multiplication.' Thus, with the above-generalized 'addition' and 'multiplication,' the set of scalars has a conceptually similar arithmetic structure as the field of reals with standard addition and multiplication underlying the linear vector spaces over which linear systems act. This alternative arithmetic structure (with operations $\vee, \star$) is minimally an idempotent semiring. Examples of 'multiplication' include the sum and the product, but $\star$ may also be only a semigroup operation. The resulting algebras include (1) the *max-plus algebra* ($\mathbb{R} \cup \{-\infty\}$, max, $+$) used in scheduling and operations research [21], discrete event systems (DES) [2,16,18,33], automated manufacturing [18,23,37], synchronization and transportation networks [2,15,30,62], max-plus control [15,18,27,30,62], optimization [2,3,15,19,28,51], geometry [19,26], morphological image analysis [31,48,59,60], and neural nets with max-plus or max–min combinations of inputs [17,56,57,66]; (2) the min-plus algebra or else known as *tropical semiring* ($\mathbb{R}\cup\{+\infty\}$, min, $+$) used in shortest paths on networks [21] and in speech recognition and natural language processing [34,52]; this is a logarithmic version of (3) the underlying *max-times* semiring ($[0, +\infty)$, $\vee$, $\times$) used for inference with belief propagation in graphical models [7,54]; (4) the *fuzzy logic or probability semiring* ($[0, 1]$, $\vee$, T) with statistical T-norms used in probabilistic automata and fuzzy neural nets [35,39], fuzzy image processing and dynamical systems [8,45,49], and fuzzy Markov chains [1]. Max-plus algebra is also a major part of *idempotent mathematics*

[41,50], a vibrant area with contributions to mathematical physics and optimization. Further, in multimodal processing for cognition modeling, several psychophysical and computational experiments indicate that the superposition of sensory signals or cognitive states seems to be better modeled using max or min rules, possibly weighted. Such an example is the work [24] on attention-based multimodal video summarization where a (possibly weighted) min/max fusion of features from the audio and visual signal channels and of salient events from various modalities seems to outperform linear fusion schemes. Tracking of these salient events was modeled in [47] using a max-product dynamical system. Finally, the problem of bridging the semantic gap in multimedia requires integration of continuous sensory modalities (like audio and/or vision) with discrete language symbols and semantics extracted from text. Similarly, in control and robotics there are efforts to develop hybrid systems that can model interactions between heterogeneous information streams like continuous inputs and symbolic strings, e.g., motion control with language-driven variables [13]. In both of these applications, we need models where the computations among modalities/states can handle both real numbers and Boolean-like variables; this is possible using max/min rules.

Motivated by the above applications, in this work we develop a theory and some tools to unify the representation and analysis of nonlinear systems whose dynamics evolve based on the following **max-**$\star$ model

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{A}(t) \circledast \boldsymbol{x}(t-1) \ \vee \ \boldsymbol{B}(t) \circledast \boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(t) \circledast \boldsymbol{x}(t) \ \vee \ \boldsymbol{D}(t) \circledast \boldsymbol{u}(t) \end{aligned} \tag{2}$$

where $\boldsymbol{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}} \in \mathcal{K}^n$ is a n-dimensional state vector with elements from the scalars' set $\mathcal{K}$, which will generally be a subset of the extended reals $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$. The linear matrix product in (1), which is based on a sum of products, is replaced in (2) by a nonlinear matrix product ($\circledast$) based on a max of $\star$ operations, where $\star$ shall denote our general scalar operation discussed in Sect. 3.1. The max-$\star$ 'multiplication' of a matrix $\boldsymbol{A} = [a_{ij}] \in \mathcal{K}^{m\times n}$ with a vector $\boldsymbol{x} = [x_i] \in \mathcal{K}^n$ yields a vector $\boldsymbol{b} = [b_i] \in \mathcal{K}^m$ defined by:

$$\boldsymbol{A} \circledast \boldsymbol{x} = \boldsymbol{b}, \quad b_i = \bigvee_{j=1}^{n} a_{ij} \star x_j \tag{3}$$

Further, the pointwise 'addition' of vectors (and possibly matrices) of same size in (1) is replaced in (2) by their pointwise $\vee$:

$$\begin{aligned} \boldsymbol{x} \vee \boldsymbol{y} &= [x_1 \vee y_1, \ldots, x_n \vee y_n]^{\mathrm{T}} \\ \boldsymbol{A} \vee \boldsymbol{B} &= [a_{ij} \vee b_{ij}] \end{aligned} \tag{4}$$

A max-plus 2×2 example of (3) is

$$\begin{bmatrix} 4 & -1 \\ 2 & -\infty \end{bmatrix} \circledast \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \quad \begin{matrix} \max(x+4, y-1) = 3 \\ x + 2 = 1 \end{matrix} \tag{5}$$

with solution $x = -1$ and $y \leq 4$.

By replacing maximum ($\vee$) with minimum ($\wedge$) and the $\star$ operation with a dual operation $\star'$ we obtain a *dual* model that describes the state-space dynamics of **min-$\star'$** systems:

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{A}(t) \boxed{\star}' \boldsymbol{x}(t-1) \ \wedge \ \boldsymbol{B}(t) \boxed{\star}' \boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C}(t) \boxed{\star}' \boldsymbol{x}(t) \ \wedge \ \boldsymbol{D}(t) \boxed{\star}' \boldsymbol{u}(t) \end{aligned} \tag{6}$$

where the min-$\star'$ matrix–vector 'multiplication' is defined by:

$$\boldsymbol{A} \boxed{\star}' \boldsymbol{x} = \boldsymbol{b}, \quad b_i = \bigwedge_{j=1}^{n} a_{ij} \star' x_j \tag{7}$$

The state equations (2) and (6) have *time-varying* coefficients. For constant matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$, we obtain the *constant-coefficient* case.

By specifying the scalar 'multiplication' $\star$ and its dual $\star'$, we obtain a large variety of classes of nonlinear dynamical systems that are described by the above unified algebraic models of the max or min type. The most well-known special case is $\star = +$, the principal interpretation of minimax algebra, which has been extensively studied in scheduling, DES, max-plus control and optimization [2,15,18,19,21,27,30,41]. In typical applications of DES in automated manufacturing, the states $x_i(t)$ represent starting times of the tth cycle of machine i, the input $\boldsymbol{u}$ represents availability times of parts, $\boldsymbol{y}$ represents completion times, and the elements of $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ represent activity durations. The homogeneous state dynamics of (2) are modeled by max-plus recursive equations:

$$x_i(t) = \max_{1 \le j \le n} a_{ij} + x_j(t-1), \quad i = 1, \ldots, n \tag{8}$$

Another special case is $\star = \times$, which is, however, less frequently related to minimax algebra and rarely viewed as a dynamical system. This is used in communications (e.g., Viterbi algorithm), in probabilistic networks [7,54] as the max-product belief propagation, and in speech recognition and language processing [34,52]. But both its max-product algebra and its general dynamics with control inputs have not been studied. Other cases are much less studied or relatively unknown.

Our theoretical analysis is based on a relatively new type of nonlinear space we have developed in recent work [46] and further refine herein, which we call *complete weighted lattice (CWL)*. This combines two vector or signal generalized 'additions' of the supremum ($\vee$) or the infimum ($\wedge$) type and two generalized scalar multiplications, $\star$ and its dual $\star'$, which distribute over $\vee$ and $\wedge$, respectively. The axioms of CWLs bear a remarkable similarity with those of linear spaces, the major difference being the lack of inverses for the sup/inf operations and sometimes for the $\star$ operation too. The present work focuses on analyzing max/min dynamical systems using CWLs, whose advantages over the minimax algebra [21], which has been so far the main algebraic framework for DES and max-plus control, include the following:

1. We believe that the theory of lattices and lattice-ordered monoids [6] offers a conceptually elegant and compact way to express the combined rich algebraic

structure instead of viewing it as a pair of two idempotent ordered semirings of minimax algebra. Although in several previous works the (max, +) and (min, +) algebras have been used at the same time, expressing and exploiting the coupling between the two becomes simpler by using the built-in duality of lattices which is at the core of their theoretical foundation.

2. Lattice monotone operators of the dilation (δ) or erosion (ε) type can be defined, as done in morphological image analysis [31,45,59], which play the role of 'linear operators' on CWLs and can represent systems obeying a max-$\star$ or a min-$\star'$ superposition, respectively. Such operators can represent both state vector transformations by matrix–vector generalized products of the max-$\star$ or min-$\star'$ type, as in (3) and (7), as well as input–output signal mappings in the form of nonlinear convolutions of two signals f and g: sup-$\star$ convolution

$$(f \circledast g)(t) \triangleq \bigvee_k f(k) \star g(t-k), \tag{9}$$

or inf-$\star'$ convolution

$$(f \circledast' g)(t) \triangleq \bigwedge_k f(k) \star' g(t-k) \tag{10}$$

The only well-known special case $\star = +$ is called supremal or infimal convolution in convex analysis and optimization [4,43,58] as well as weighted (Minkowski) signal dilation or erosion in morphological image analysis and vision [31,48,59, 60]. Other cases are much less studied or relatively unknown.

3. Modeling the information flow in these dynamical systems via the above lattice operators is greatly enabled by the concept of *adjunction*, which is a pair (ε, δ) of erosion and dilation operators forming a type of duality expressed by the following

$$\delta(\boldsymbol{x}) \leq \boldsymbol{y} \Longleftrightarrow \boldsymbol{x} \leq \varepsilon(\boldsymbol{y}) \tag{11}$$

for any vectors or signals $\boldsymbol{x}$, $\boldsymbol{y}$.

From a geometrical viewpoint, we may call the CWLs *polygonal spaces* because of the geometric shape of the corner-forming piecewise-straight lines $y = \max(a+x, b)$ or $y = \max(ax, b)$ and their duals (by replacing max with min) which express the basic algebraic superpositions in CWLs, in analogy to the geometry of the straight line $y = ax + b$ which expresses in a simplified way the basic superposition in linear spaces. See Fig. 1 for an example.

Contributions of our work:

(i) Unify all types of max-$\star$ and min-$\star'$ systems under a common theoretical framework of *complete weighted lattices* (CWLs). Further, while previous work focused mainly on the (max, +) or (min, +) formalism, we join both using CWLs and generalize them by replacing + with any operation $\star$ that distributes over $\vee$ and a dual operation $\star'$ that distributes over $\wedge$. The corresponding generalized scalar arithmetic is governed by a rich algebraic structure, called *clodum*,

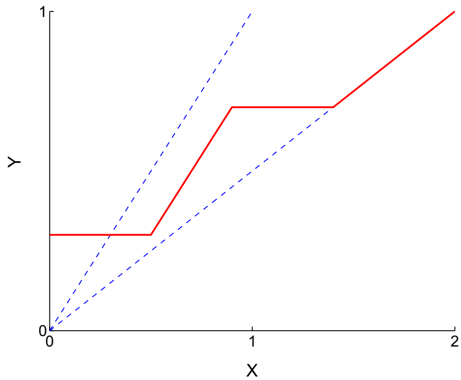

Fig. 1 Geometry of basic superposition via a straight line (dashed) in linear spaces versus the polygonal line (solid line) in complete weighted lattices—the *polygonal spaces*. The polygonal line is $y = \min[\max(x - 0.2, 0.3), \max(x/2, 0.7)]$

which we developed in previous work [45,46] and further refine herein. This clodum serves as the 'field of scalars' for the CWLs and binds together a pair of dual 'additions' with a pair of dual 'multiplications'; as opposed to max-plus, in some cases the 'multiplications' do not have inverses. Two examples different from max-plus, which we analyze in some detail with applications, are the max-product and the max–min cases.

(ii) Analyze the nonlinear system dynamics both in state space using a CWL matrix–vector algebra and in the input–output signal space using sup/inf-$\star$ convolutions, represented via lattice monotone operators in adjunction pairs. For the above, we have used the common formalism of CWLs to model both finite- and infinite-dimensional spaces.

(iii) Enable and simplify the analysis and proofs of various results in system representation using lattice adjunctions. Further, use the latter to generate lattice projections that provide optimal solutions for max-$\star$ equations $\boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{b}$. Since the constituent operators of the lattice adjunctions are dilations and erosions which have a geometrical interpretation and have found numerous applications in image analysis, the above perspective to nonlinear system analysis also offers some geometrical insights.

(iv) Study causality, stability, and controllability of max-$\star$ and min-$\star'$ systems and link stability with spectral analysis in max-$\star$ algebra and controllability with lattice projections.

(v) Advance the study of special cases employed in many application areas: (a) Nonlinear systems represented by max/min-sum ($\star = +$) difference equations, as applied to geometric filtering and shortest path computation. State equations and stability analysis of recursive nonlinear filters. (b) Max-product systems ($\star = \times$) that extend the Viterbi algorithm of hidden Markov models to cases with control inputs and can model cognitive processes related to audiovisual attention. (c) Probabilistic automata and fuzzy Markov chains governed by max/min rules and with arithmetic based on triangular norms.

Notation: We think that the currently used notation in max-plus algebra of $\oplus$ and $\otimes$ to denote the maximum $\vee$ ('addition') and the $\star$ ('multiplication'), respectively,

Table 1 Notation for the main algebraic operations

Operation	Meaning
$\bigvee$	Maximum/supremum: applies for scalars, vectors and matrices
$\bigwedge$	Minimum/infimum: applies for scalars, vectors and matrices
$\boxed{\star}$ ($\boxed{\star}'$)	General max-$\star$ (min-$\star'$) matrix multiplication
$\boxplus$ ($\boxplus'$)	Max-sum (min-sum) matrix multiplication
$\boxtimes$ ($\boxtimes'$)	Max-product (min-product) matrix multiplication
$\circledast$($\circledast'$)	General max-$\star$ (min-$\star'$) signal convolution
$\oplus$ ($\oplus'$)	Max-sum (min-sum) signal convolution
$\otimes$ ($\otimes'$)	Max-product (min-product) signal convolution

obscures the lattice operations; in contrast, our proposed notation is simpler and more realistic since it uses the well-established symbols $\vee$, $\wedge$ for sup/inf operations and does not bias the arbitrary scalar binary operation $\star$ with the symbol $\otimes$. Further, the symbol $\oplus$ has been extensively used in signal and image processing for the max-plus signal convolution; herein, we continue this notation. Table 1 summarizes the main symbols of our notation. We use roman letters for functions, signals and their arguments and Greek letters for operators and also boldface roman letters for vectors (lower case) and matrices (capital). If $\boldsymbol{M} = [m_{ij}]$ is a matrix, its (i, j)th element is also denoted as $\{\boldsymbol{M}\}_{ij} = m_{ij}$. Similarly, $\boldsymbol{x} = [x_i]$ denotes a column vector, whose ith element is denoted as $\{\boldsymbol{x}\}_i$ or simply x_i.

2 Lattices and monotone operators

Most of the background material in this section follows [6,31,32,46,59].

2.1 Lattices

A partially ordered set, briefly *poset* $(\mathcal{P}, \leq)$, is a set $\mathcal{P}$ with a binary relation $\leq$ that is a *partial ordering*, i.e., is reflexive, antisymmetric and transitive. If, in addition, for any two elements $X, Y \in \mathcal{P}$ we have either $X \leq Y$ or $Y \leq X$, then $\mathcal{P}$ is called a *chain*. To every partial ordering $\leq$, there corresponds a *dual partial ordering* $\leq'$ defined by '$X \leq' Y$ iff $X \geq Y$.' Let $\mathcal{S}$ be a subset of $(\mathcal{P}, \leq)$; an upper bound of $\mathcal{S}$ is an element $B \in \mathcal{P}$ such that $X \leq B$ for all $X \in \mathcal{S}$. The least upper bound of $\mathcal{S}$ is called its **supremum** and denoted by $\sup \mathcal{S}$ or $\bigvee \mathcal{S}$. By duality, we define the greatest lower bound of $\mathcal{S}$, called its **infimum** and denoted by $\inf \mathcal{S}$ or $\bigwedge \mathcal{S}$. If the supremum (resp. infimum) of $\mathcal{S}$ belongs to $\mathcal{S}$, then it is called the *greatest element* or *maximum* (resp. *least element* or *minimum*) of $\mathcal{S}$. An element M of $\mathcal{S}$ is called *maximal* (resp. *minimal*) if there is no other element in $\mathcal{S}$ that is greater (resp. smaller) than M.

A **lattice** is a poset $(\mathcal{L}, \leq)$ any two of whose elements have a supremum, denoted by $X \vee Y$, and an infimum, denoted by $X \wedge Y$. We often denote the lattice structure by $(\mathcal{L}, \vee, \wedge)$. A lattice $\mathcal{L}$ is *complete* if each of its subsets (finite or infinite) has a supremum and an infimum in $\mathcal{L}$. Any nonempty complete lattice is universally bounded because it contains its supremum $\top = \bigvee \mathcal{L}$ and infimum $\bot = \bigwedge \mathcal{L}$ which are its greatest (top) and least (bottom) elements, respectively. In any lattice $\mathcal{L}$, by replacing the partial ordering with its dual and by interchanging the roles of the supremum and infimum we obtain a *dual lattice* $\mathcal{L}'$. *Duality principle*: to every definition, property and statement that applies to $\mathcal{L}$ there corresponds a dual one that applies to $\mathcal{L}'$ by interchanging $\leq$ with $\leq'$ and $\vee$ with $\wedge$. A bijection between two lattices $\mathcal{L}$ and $\mathcal{M}$ is called an *isomorphism* (resp. dual isomorphism) if it preserves (resp. reverses) suprema and infima. If $\mathcal{L} = \mathcal{M}$, a (dual-) isomorphism on $\mathcal{L}$ is called *(dual-)automorphism*.

The lattice operations satisfy many properties, as summarized in Table 3. Conversely, a set $\mathcal{L}$ equipped with two binary operations $\vee$ and $\wedge$ that satisfy properties (L1, L1$'$)–(L5, L5$'$) is a lattice whose supremum is $\vee$, the infimum is $\wedge$, and partial ordering $\leq$ is given by (L6). A lattice $(\mathcal{L}, \vee, \wedge)$ contains two weaker substructures: a sup-semilattice $(\mathcal{L}, \vee)$ that satisfies properties (L1–L4) and an inf-semilattice $(\mathcal{L}, \wedge)$ that satisfies properties (L1$'$–L4$'$).

The additional properties (L7, L7$'$) and (L8, L8$'$) in Table 3 hold only if the lattice contains a least and a greatest element, respectively. A lattice $\mathcal{L}$ is called *distributive* if it satisfies properties (L9, L9$'$); if these also hold over infinite set collections, then the lattice is called *infinitely distributive*. The rest of the properties of Table 3, labeled as 'WL#,' refer to a richer algebra defined as 'weighted lattices' in Sect. 3.2.

Examples 1 (a) Any *chain* is an infinitely distributive lattice. Thus, the chain $(\mathbb{R}, \leq)$ of real numbers equipped with the natural order $\leq$ is a lattice, but not complete. The set of extended real numbers $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ is a complete lattice.

(b) The *power set* $\mathcal{P}(E) = \{X : X \subseteq E\}$ of an arbitrary set E equipped with the partial order of set inclusion is an infinitely distributive lattice under the supremum and infimum induced by set inclusion which are the set union and intersection, respectively.

(c) In a lattice $\mathcal{L}$ with universal bounds $\bot$ and $\top$, an element $X \in \mathcal{L}$ is said to have a *complement* $X^c \in \mathcal{L}$ if $X \vee X^c = \top$ and $X \wedge X^c = \bot$. If all the elements of $\mathcal{L}$ have complements, then $\mathcal{L}$ is called complemented. Any complemented and distributive lattice $\mathcal{B}$ is called a *Boolean lattice*.

(d) Let $\mathcal{L}^E = \mathrm{Fun}(E, \mathcal{L})$ denote the set of all functions $f : E \to \mathcal{L}$. If $\leq$ is the partial ordering of $\mathcal{L}$, we can equip the function space $\mathcal{L}^E$ with the pointwise partial ordering $f \leq g$, which means $f(x) \leq g(x)\ \forall x \in E$, the pointwise supremum $(\bigvee_i f_i)(x) = \bigvee_i f_i(x)$, and pointwise infimum $(\bigwedge_i f_i)(x) = \bigwedge_i f_i(x)$. Then, $(\mathcal{L}^E, \vee, \wedge)$ becomes a *function lattice*, which retains possible properties of $\mathcal{L}$ of being complete, or (infinitely) distributive, or Boolean.

2.2 Operators on lattices

Let $\mathcal{O}(\mathcal{L})$ be the set of all *operators* on a complete lattice $\mathcal{L}$, i.e., mappings from $\mathcal{L}$ to itself. Given two such operators ψ and ϕ, we can define a partial ordering between them

($\phi \le \psi$), their supremum ($\psi \vee \phi$) and infimum ($\psi \wedge \phi$) in a pointwise way, as done in Example 1(d). This makes $\mathcal{O}(\mathcal{L})$ a complete function lattice which inherits many of the possible properties of $\mathcal{L}$. Further, we define the composition of two operators as an operator product: $\psi\phi(X) \triangleq \psi(\phi(X))$; special cases are the operator powers $\psi^n = \psi\psi^{n-1}$. Some useful types and properties of lattice operators ψ include the following: (i) identity: $\mathbf{id}(X) = X \ \ \forall X \in \mathcal{L}$. (ii) extensive: $\psi \ge \mathbf{id}$. (iii) antiextensive: $\psi \le \mathbf{id}$. (iv) idempotent: $\psi^2 = \psi$. (v) involution: $\psi^2 = \mathbf{id}$.

2.2.1 Monotone operators

Of great interest are the monotone operators, whose collections form sublattices of $\mathcal{O}(\mathcal{L})$. They come in three basic kinds according to which of the following lattice structures they preserve (or map to its dual): (i) partial ordering, (ii) supremum, (iii) infimum.

A lattice operator ψ is called *increasing* or *isotone* if it is order-preserving, i.e., $X \le Y \Longrightarrow \psi(X) \le \psi(Y)$. A lattice operator ψ is called *decreasing* or *antitone* if it is order-inverting, i.e., $X \le Y \Longrightarrow \psi(X) \ge \psi(Y)$.

Examples of increasing operators are the lattice homomorphisms which preserve suprema and infima over finite collections. If a lattice homomorphism is also a bijection, then it becomes an automorphism. A bijection ϕ is an automorphism if both ϕ and its inverse ϕ^{-1} are increasing.

Four types of increasing operators, fundamental for unifying systems on lattices, are the following:

$$\begin{aligned}
\delta \text{ is } \textit{dilation} &\text{ iff } \delta(\textstyle\bigvee_{i \in J} X_i) = \bigvee_{i \in J} \delta(X_i) \\
\varepsilon \text{ is } \textit{erosion} &\text{ iff } \varepsilon(\textstyle\bigwedge_{i \in J} X_i) = \bigwedge_{i \in J} \varepsilon(X_i) \\
\alpha \text{ is } \textit{opening} &\text{ iff } \text{increasing, idempotent, antiextensive} \\
\beta \text{ is } \textit{closing} &\text{ iff } \text{increasing, idempotent, extensive}
\end{aligned}$$

Dilations and erosions require arbitrary (possibly infinite) collections $\{X_i : i \in J\}$ of lattice elements; hence, they need complete lattices. The special case of an empty collection equips each dilation and erosion with the following necessary properties:

$$\delta(\bot) = \bot, \quad \varepsilon(\top) = \top \tag{12}$$

The four above types of lattice operators were originally defined in [31,59] as generalizations of the corresponding Minkowski-type morphological operators and have been applied in numerous image processing tasks.

Examples of decreasing operators are the dual homomorphisms, which interchange suprema with infima. A lattice dual-automorphism is a bijection θ that interchanges suprema with infima, or equivalently iff it is a bijection and both θ and its inverse θ^{-1} are decreasing. A *negation* ν is a dual-automorphism that is also involutive; we may write $X^{\neg} = \nu(X)$ for the negative of a lattice element. Given an operator ψ in a lattice equipped with a negation, its corresponding *negative* (a.k.a. *dual*) operator is defined as $\psi^{\neg}(X) \triangleq [\psi(X^{\neg})]^{\neg}$. For example, the most well-known negation on the set lattice $\mathcal{P}(E)$ is the complementation $\nu(X) = X^c = E \backslash X$, whereas on the function lattice $\text{Fun}(E, \overline{\mathbb{R}})$ the most well-known negation is $\nu(f) = -f$.

The above definitions allow broad classes of operators on vector or signal spaces to be grouped as parallel or sequential combinations of lattice monotone operators and their common properties to be studied under the unifying lattice framework. In this work, we shall find them very useful for representing the state and output responses or for approximating solutions of systems obeying a supremal or infimal superposition.

2.2.2 Order continuity

Consider an arbitrary sequence (X_n) of elements in a complete lattice $\mathcal{L}$. The following two limits can be defined using only sup/inf combinations:

$$\limsup X_n \triangleq \bigwedge_{n\geq 1} \bigvee_{k\geq n} X_k \quad \liminf X_n \triangleq \bigvee_{n\geq 1} \bigwedge_{k\geq n} X_k \tag{13}$$

In general, $\liminf X_n \leq \limsup X_n$. A sequence (X_n) is defined to *order converge* to a lattice element X, written as $X_n \overset{\text{ord}}{\to} X$, if $\liminf X_n = \limsup X_n = X$.

An operator ψ on $\mathcal{L}$ is called $\downarrow$-*continuous* if $(X_n) \overset{\text{ord}}{\to} X$ in $\mathcal{L}$ implies that $\limsup \psi(X_n) \leq \psi(X)$. Dually, ψ is called $\uparrow$-*continuous* if $(X_n) \overset{\text{ord}}{\to} X$ implies that $\liminf \psi(X_n) \geq \psi(X)$. Finally, ψ is called *order continuous* if it is both $\downarrow$-continuous and $\uparrow$-continuous. On a chain, e.g., $(\overline{\mathbb{R}}, \leq)$, the concepts of order convergence and order continuity coincide with their topological counterparts.

There is a stronger form of order convergence applicable to monotone sequences. We write $X_n \downarrow X$ to mean a *monotonic convergence* where (X_n) is a decreasing sequence $(X_{n+1} \leq X_n)$ and $X = \bigwedge_n X_n$. Dually, we write $X_n \uparrow X$ to mean that (X_n) is an increasing sequence $(X_{n+1} \geq X_n)$ and $X = \bigvee_n X_n$. This monotonic convergence allows to easily examine the order continuity of increasing operators. Specifically, an increasing operator ψ on a complete lattice $\mathcal{L}$ is $\downarrow$-continuous iff $X_n \downarrow X$ implies that $\psi(X_n) \downarrow \psi(X)$ for any sequence (X_n). Dually, ψ is $\uparrow$-continuous iff $X_n \uparrow X$ implies that $\psi(X_n) \uparrow \psi(X)$. This result implies that, since dilations (resp. erosions) distribute over arbitrary suprema (resp. infima), dilations are $\uparrow$-continuous, whereas erosions are $\downarrow$-continuous.

2.2.3 Residuation and adjunctions

An increasing operator ψ on a complete lattice $\mathcal{L}$ is called **residuated** [9,10] if there exists an increasing operator $\psi^\sharp$ such that

$$\psi\psi^\sharp \leq \mathbf{id} \leq \psi^\sharp\psi \tag{14}$$

$\psi^\sharp$ is called the **residual** of ψ and is the closest to being an inverse of ψ. Specifically, the residuation pair $(\psi, \psi^\sharp)$ can solve inverse problems of the type $\psi(X) = Y$ either exactly since $\hat{X} = \psi^\sharp(Y)$ is the greatest solution of $\psi(X) = Y$ if a solution exists, or approximately since $\hat{X}$ is the greatest *subsolution* in the sense that $\hat{X} = \bigvee\{X :$

$\psi(X) \le Y\}$. On complete lattices an increasing operator ψ is residuated (resp. a residual $\psi^\sharp$) if and only if it is a dilation (resp. erosion). The residuation theory has been used for solving inverse problems in matrix algebra [2,19,21] over the max-plus or other idempotent semirings.

Dilations and erosions come in pairs as the following concept reveals. The pair (δ, ε) of operators on a complete lattice $\mathcal{L}$ is called an **adjunction**[1] on $\mathcal{L}$ if

$$\delta(X) \le Y \iff X \le \varepsilon(Y) \quad \forall X, Y \in \mathcal{L} \tag{15}$$

In any adjunction, (15) implies that δ is a dilation and ε is an erosion. It can be shown that this double inequality is equivalent to the inequality (14) satisfied by a residuation pair of increasing operators if we identify the residuated map ψ with δ and its residual $\psi^\sharp$ with ε. To view (δ, ε) as an adjunction instead of a residuation pair has the advantage of the additional geometrical intuition and visualization afforded by the dilation and erosion operators, which are well known in image analysis and can be interpreted as augmentation and shrinkage, respectively, of input sets or of hypographs of functions.

In any adjunction (δ, ε), ε is called the *adjoint erosion* of δ, whereas δ is the *adjoint dilation* of ε. There is a one-to-one correspondence between the two operators of an adjunction pair, since, given a dilation δ, there is a unique erosion

$$\varepsilon(Y) = \bigvee\{X \in \mathcal{L} : \delta(X) \le Y\} \tag{16}$$

such that (δ, ε) is adjunction. Conversely, given an erosion ε, there is a unique dilation

$$\delta(X) = \bigwedge\{Y \in \mathcal{L} : \varepsilon(Y) \ge X\} \tag{17}$$

such that (δ, ε) is adjunction. Adjunctions create operator duality pairs that are different than negation in the sense that one operator is the closest to being the inverse of the other, either from below or above.

2.2.4 Projections on lattices

A large variety of useful lattice operators share two properties: *increasing and idempotent*. Such operators were called *morphological filters* in [31,59]. We shall call them *lattice projections* of the order type, since they preserve the lattice ordering and are idempotent in analogy with the linear projections that preserve the algebraic structure of linear spaces and are idempotent. Two well-studied special cases of lattice projections are the openings and closings, each of which has an additional property. Specifically, openings are lattice projections that are antiextensive, whereas closings are extensive projections. Combinations of such generalized filters have proven to be very useful for signal denoising, image enhancement, simplification, segmentation, and object detection. From the composition of the erosion and dilation of any

[1] As explained in [31,32], the adjunction is related to a concept in poset and lattice theory called '*Galois connection*.' In [31,59] an adjunction pair is denoted as (ε, δ), but in this paper we prefer to reverse the positions of its two operators, so that it agrees with the structure of a residuation pair $(\psi, \psi^\sharp)$.

adjunction (δ, ε), we can generate a projection $\alpha = \delta\varepsilon$ that is also an opening since $\alpha(X) \leq X$ and $\alpha^2 = \alpha$. To prove this note that, by (15),

$$\delta\varepsilon \leq \mathbf{id} \leq \varepsilon\delta \tag{18}$$

which implies that $\delta\varepsilon\delta\varepsilon = \delta\varepsilon$. Dually, any adjunction can also generate a closing projection $\beta = \varepsilon\delta$, which always satisfies $\beta(X) \geq X$ and $\beta^2 = \beta$. There are also other types of lattice projections that are studied in [19].

2.3 Lattice-ordered monoids and clodum

A lattice $(\mathcal{M}, \vee, \wedge)$ is often endowed with a third binary operation, called symbolically the 'multiplication' $\star$, under which $(\mathcal{M}, \star)$ is a group or monoid or just semigroup [6].

Consider now an algebra $(\mathcal{M}, \vee, \wedge, \star, \star')$ with four binary operations, which we call a *lattice-ordered double monoid*, where $(\mathcal{M}, \vee, \wedge)$ is a lattice, $(\mathcal{M}, \star)$ is a monoid whose 'multiplication' $\star$ distributes over $\vee$, and $(\mathcal{M}, \star')$ is a monoid whose 'multiplication' $\star'$ distributes over $\wedge$. These distributivities imply that both $\star$ and $\star'$ are increasing. To the above definitions, we add the word *complete* if $\mathcal{M}$ is a complete lattice and the distributivities involved are infinite. We call the resulting algebra a *complete lattice-ordered double monoid*, in short *clodum* [45,46].

Previous works on minimax or max-plus algebra and their applications have used alternative names[2] for algebraic structures similar to the above definitions which emphasize semigroups and semirings instead of lattices. If $\star = \star'$, we have a self-dual 'multiplication.' This always happens if $(\mathcal{M}, \star)$ is a group, i.e., a monoid where each element has an inverse; in this case we obtain a *lattice-ordered group*, and the group 'multiplication' $X \mapsto A \star X$ is a lattice automorphism.

We give a precise definition of a general clodum and some examples since this will be one of the fundamental algebraic structures to build the nonlinear spaces in our work. An algebraic structure $(\mathcal{K}, \vee, \wedge, \star, \star')$ is called a **clodum** if:

(C1) $(\mathcal{K}, \vee, \wedge)$ is a complete distributive lattice.
(C2) $(\mathcal{K}, \star)$ is a monoid whose operation $\star$ is a dilation.
(C3) $(\mathcal{K}, \star')$ is a monoid whose operation $\star'$ is an erosion.

Remarks

(i) As a lattice, $\mathcal{K}$ is not necessarily infinitely distributive, although in this paper all our examples will be such.

[2] Minimax algebra [21] has been based on bands (idempotent semigroups) and belts (idempotent presemirings), whereas max-plus algebra and its application to DES [2,15,20,28] is based on dioids (canonically ordered semirings). In [21], a semilattice is called a *commutative band* and a lattice is called *band with duality*. Further, a *belt* is a semilattice-ordered semigroup, and a *belt with duality* [21] is a pair of two idempotent predioids [28] whose 'additions' are dual and form a lattice. Adding to a belt $(\mathcal{B}, \vee, \star)$ identity elements for $\star$ and $\vee$, the latter of which is also an absorbing null for $\star$, creates an idempotent *dioid* [2,20,28]. More general (including nonidempotent) dioids are studied in [28]. Finally, belts that are groups under the 'multiplication' $\star$ and as lattices have global bounds are called *blogs* (bounded lattice-ordered groups) in [21].

(ii) The clodum 'multiplications' $\star$ and $\star'$ do not have to be commutative.
(iii) The least (greatest) element $\perp$ ($\top$) of $\mathcal{K}$ is both the identity element for $\vee$ ($\wedge$) and an absorbing null for $\star$ ($\star'$) due to (12).

If $\star = \star'$ over $G = \mathcal{K} \backslash \{\perp, \top\}$ where $(G, \star)$ is a group and $(G, \vee, \wedge)$ a conditionally complete lattice, then the clodum $\mathcal{K}$ becomes a richer structure which we call a *complete lattice-ordered group*, in short **clog**. By extending properties of lattice-ordered groups [6] to clogs, we can show that in any clog the distributivity between $\vee$ and $\wedge$ is of the infinite type and the 'multiplications' $\star$ and $\star'$ are commutative. Thus, a clog has a richer structure than a *blog* (bounded lattice-ordered group) as defined in [21], because a clog is a complete and commutative blog.

Examples 2 (a) Our scalar arithmetic in this paper will use a numeric commutative clodum. Two such examples follow:
(a1) Max-plus clog[3]: $(\overline{\mathbb{R}}, \vee, \wedge, +, +')$, where $\vee/\wedge$ denote the standard sup/inf on $\overline{\mathbb{R}}$, $+$ is the standard addition on the set $\overline{\mathbb{R}}$ of extended reals playing the role of a 'multiplication' $\star$ with $+'$ being the 'dual multiplication' $\star'$; the operations $+$ and $+'$ are identical for finite reals, but $a + (-\infty) = -\infty$ and $a +' (+\infty) = +\infty$ for all $a \in \overline{\mathbb{R}}$.
(a2) Max–min clodum: $([0, 1], \vee, \wedge, \min, \max)$, where $\star = \min$ and $\star' = \max$.

(b) Matrix max-sum clodum: $(\overline{\mathbb{R}}^{n \times n}, \vee, \wedge, \boxplus, \boxplus')$ where $\overline{\mathbb{R}}^{n \times n}$ is the set of $n \times n$ matrices with entries from $\overline{\mathbb{R}}$, $\vee$ and $\wedge$ denote here elementwise matrix supremum and infimum, and $\boxplus$, $\boxplus'$ denote max-sum and min-sum matrix 'multiplications':

$$\boldsymbol{C} = \boldsymbol{A} \boxplus \boldsymbol{B} = [c_{ij}], \quad c_{ij} = \bigvee_{k=1}^{n} a_{ik} + b_{kj} \tag{19}$$

$$\boldsymbol{D} = \boldsymbol{A} \boxplus' \boldsymbol{B} = [d_{ij}], \quad d_{ij} = \bigwedge_{k=1}^{n} a_{ik} +' b_{kj} \tag{20}$$

This is a clodum with noncommutative 'multiplications.'

3 Representations of vector and signal operations on weighted lattices

3.1 Algebraic structures on the scalars

We assume that all elements of the vectors, matrices, or signals involved in the description of the systems examined herein take their values from a set $\mathcal{K}$ of *scalars*, which in general will be a subset of $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, \infty\}$ with the natural ordering $\leq$ of extended real numbers. We assume that the chain $(\mathcal{K}, \leq)$ is universally bounded, i.e., contains its least $\perp \triangleq \bigwedge \mathcal{K}$ and greatest element $\top \triangleq \bigvee \mathcal{K}$. For the weighted lattice model, we need to equip $\mathcal{K}$ with four binary operations:

[3] In every clodum and clog, we have a pair of dual 'additions' and a pair of dual 'multiplications.' However, for brevity, we assign them shorter names that contain only one 'addition' (max) and one 'multiplication,' e.g., 'max-plus clog'.

(A). The standard maximum or supremum $\vee$ on $\overline{\mathbb{R}}$, which plays the role of a generalized '*addition*.'

(A$'$). The standard minimum or infimum $\wedge$ on $\overline{\mathbb{R}}$, which plays the role of a generalized '*dual addition*.'

(M). A commutative generalized '*multiplication*' $\star$ under which: (i) $\mathcal{K}$ is a monoid with ('unit') identity element e and ('zero') null element $\perp$, i.e.,

$$a \star e = a, \quad a \star \perp = \perp, \quad \forall a \in \mathcal{K}, \tag{21}$$

and (ii) $\star$ is a scalar dilation, i.e., distributes over any supremum:

$$a \star \left(\bigvee_i x_i \right) = \bigvee_i a \star x_i \tag{22}$$

(M$'$). A commutative '*dual*[4] *multiplication*' $\star'$ under which: (i) $\mathcal{K}$ is a monoid with identity e' and null element $\top$, i.e.,

$$a \star' e' = a, \quad a \star' \top = \top, \quad \forall a \in \mathcal{K}, \tag{23}$$

and (ii) $\star'$ is a scalar erosion, i.e., distributes over any infimum:

$$a \star' \left(\bigwedge_i x_i \right) = \bigwedge_i a \star' x_i \tag{24}$$

Under the above assumptions $(\mathcal{K}, \vee, \wedge, \star, \star')$ becomes a **scalar clodum**. Note that, in addition to the minimal requirements of a general clodum in Sect. 2.3, we assume *commutative* operations $\star, \star'$. Further, the rich structure of $\overline{\mathbb{R}}$ endows the set $\mathcal{K}$ to be infinitely distributive as a lattice. This will be the most general and minimally required algebraic structure we consider for the set of scalars. We avoid degenerate cases by assuming that $\vee \neq \star$ and $\wedge \neq \star'$. However, $\star$ may be the same as $\star'$, in which case we have a self-dual 'multiplication.'

A clodum $\mathcal{K}$ is called *self-conjugate* if it has a lattice negation (i.e., involutive dual-automorphism) that maps each element a to its *conjugate* a^* such that

$$\left(\bigvee_i a_i \right)^* = \bigwedge_i a_i{}^*, \quad \left(\bigwedge_i b_i \right)^* = \bigvee_i b_i{}^*, \quad (a \star b)^* = a^* \star' b^* \tag{25}$$

We assume that the suprema and infima in (25) may be over any (possibly infinite) collections.

The set of scalars can be partitioned as $\mathcal{K} = G \cup \{\perp, \top\}$; the members of G are called the *finite scalars*, borrowing terminology from the case when $\mathcal{K} = \overline{\mathbb{R}}$. This is

[4] It is simply a matter a convention that we selected to call $\wedge$ and $\star'$ as '*dual* addition and multiplication' (instead of $\vee$ and $\star$).

Table 2 Scalar arithmetic in a CLOG

$a \in \mathcal{K}$	$b \in \mathcal{K}$	$\vee$	$\wedge$	$\star$	$\star'$
$\bot$	$\bot$	$\bot$	$\bot$	$\bot$	$\bot$
$\bot$	$y \in G$	y	$\bot$	$\bot$	$\bot$
$\bot$	$\top$	$\top$	$\bot$	$\bot$	$\top$
$x \in G$	$y \in G$	$x \vee y$	$x \wedge y$	$x \star y$	$x \star' y$
$\top$	$\bot$	$\top$	$\bot$	$\bot$	$\top$
$\top$	$y \in G$	$\top$	y	$\top$	$\top$
$\top$	$\top$	$\top$	$\top$	$\top$	$\top$

useful for cases where $(G, \star)$ is a commutative *group*. Then, for each $a \in G$ there exists its 'multiplicative inverse' a^{-1} such that $a \star a^{-1} = e$. Further, the 'multiplication' $\star$ and its self-dual $\star'$ (which coincide over G) can be extended over the whole $\mathcal{K}$ by adding the rules in (21) and (23) involving the null elements. As defined in Sect. 2.3, the resulting richer structure is a **clog**. Whenever $\mathcal{K}$ is a clog, it becomes self-conjugate by setting

$$a^* = \begin{cases} a^{-1} & \text{if } \bot < a < \top \\ \top & \text{if } a = \bot \\ \bot & \text{if } a = \top \end{cases} \tag{26}$$

Next we further elaborate on three main examples used in this paper for a scalar clodum.

Examples 3

(a) *Max-plus* clog $(\overline{\mathbb{R}}, \vee, \wedge, +, +')$: This is the archetypal example of a clog. The identities are $e = e' = 0$, the nulls are $\bot = -\infty$ and $\top = +\infty$, and the conjugation mapping is $a^* = -a$.

(b) *Max-times* clog $([0, +\infty], \vee, \wedge, \times, \times')$: The identities are $e = e' = 1$, the nulls are $\bot = 0$ and $\top = +\infty$, and the conjugation mapping is $a^* = 1/a$.

(c) *Max–min* clodum $([0, 1], \vee, \wedge, \min, \max)$: As 'multiplications' we have $\star = \min$ and $\star' = \max$. The identities and nulls are $e' = \bot = 0$, $e = \top = 1$. A possible conjugation mapping is $a^* = 1 - a$. Additional cloda that are not clogs are discussed in Sect. 9.3 using more general fuzzy intersections and unions.

Table 2 summarizes the results of all scalar binary operations in a clog. We see that in a clog the $\star$ and $\star'$ coincide in all cases with only one exception: the combination of the least and greatest elements. Henceforth when a clodum $\mathcal{K}$ is a clog we can denote the algebra as $(\mathcal{K}, \vee, \wedge, \star)$ using only one 'multiplication' operation and the case $\bot \star \top$ will have value $\bot$ (resp. $\top$) if it is combined with other terms via a supremum (resp. infimum).

3.2 Complete weighted lattices

Consider a nonempty collection $\mathcal{W}$ of mathematical objects, which will be our space; examples of such objects include the vectors in $\overline{\mathbb{R}}^n$ or signals in $\mathrm{Fun}(E, \overline{\mathbb{R}})$. Thus, we shall symbolically refer to the space elements as 'vectors/signals,' although they may be arbitrary objects. Also, consider a clodum $(\mathcal{K}, \vee, \wedge, \star, \star')$ of 'scalars.'[5] We define two internal operations among vectors/signals X, Y in $\mathcal{W}$: their supremum $X \vee Y : \mathcal{W}^2 \to \mathcal{W}$ and infimum $X \wedge Y : \mathcal{W}^2 \to \mathcal{W}$, which we denote using the same supremum symbol ($\vee$) and infimum symbol ($\wedge$) as in the clodum, hoping that the differences will be clear to the reader from the context. Further, we define two external operations among any vector/signal X in $\mathcal{W}$ and any scalar c in $\mathcal{K}$: a 'scalar multiplication' $c \star X : (\mathcal{K}, \mathcal{W}) \to \mathcal{W}$ and a 'scalar dual multiplication' $c \star' X : (\mathcal{K}, \mathcal{W}) \to \mathcal{W}$, again by using the same symbols as in the clodum. Now, we define $\mathcal{W}$ to be a **weighted lattice** space over the clodum $\mathcal{K}$ if for all $X, Y, Z \in \mathcal{W}$ and $a, b \in \mathcal{K}$ all the axioms of Table 3 hold. Note[6] that: (a) Under axioms L1–L9 and their duals L1′–L9′, $\mathcal{W}$ is a distributive lattice with a least element (O) and a greatest element (I). (b) These axioms bear a striking similarity with those of a linear space. One difference is that the vector/signal addition ($+$) of linear spaces is now replaced by two dual superpositions, the lattice supremum ($\vee$) and infimum ($\wedge$); further, the scalar multiplication ($\times$) of linear spaces is now replaced by two operations $\star$ and $\star'$ that are dual to each other. Only one major property of the linear spaces is missing from the weighted lattices: the existence of 'additive inverses'; i.e., the supremum and infimum operations do not have inverses.

We define the weighted lattice $\mathcal{W}$ to be a **complete weighted lattice (CWL)** space if all the following hold:

(i) $\mathcal{W}$ is closed under any, possibly infinite, suprema and infima.
(ii) The distributivity laws between the scalar operations $\star$ ($\star'$) and the supremum (infimum) are of the infinite type.

Note that, a clodum is by itself a complete weighted lattice over itself.

Consider a subset $\mathcal{A}$ of a complete weighted lattice $\mathcal{W}$ over a clodum $\mathcal{K}$. A space element F is called a **sup-$\star$ combination** of points in $\mathcal{A}$ if there exists an indexed set of space elements $\{F_i\}$ in $\mathcal{A}$ and a corresponding set of scalars $\{a_i\}$ in $\mathcal{K}$ such that

$$F = \bigvee_i a_i \star F_i, \tag{27}$$

Dually, we can form an **inf-$\star'$ combination** $G = \bigwedge_i b_i \star' G_i$ of points G_i in $\mathcal{A}$ with scalars b_i. The *sup-$\star$ span* of $\mathcal{A}$, denoted by $\mathrm{span}_\vee(\mathcal{A})$, is the set of all sup-$\star$ combinations of elements in $\mathcal{A}$. If $\mathcal{A} = \varnothing$, then $\mathrm{span}_\vee(\mathcal{A}) = \{O\}$. Dually, the set of

[5] In this paper, as 'scalars' we use numbers from $\overline{\mathbb{R}}$ or its subsets, but the general definition of a weighted lattice allows for an arbitrary clodum as the set of 'scalars.'

[6] If in our definition of a weighted lattice, one focuses only on one vector 'addition,' say the vector supremum, and its corresponding scalar 'multiplication,' then the weaker algebraic structure becomes an idempotent semimodule over an idempotent semiring. This has been studied in [19,28,41].

Table 3 Axioms of weighted lattices

Sup-semilattice	Inf-semilattice	Description
L1. $X \vee Y \in \mathcal{W}$	L1′. $X \wedge Y \in \mathcal{W}$	Closure of $\vee, \wedge$
L2. $X \vee X = X$	L2′. $X \wedge X = X$	Idempotence of $\vee, \wedge$
L3. $X \vee Y = Y \vee X$	L3′. $X \wedge Y = Y \wedge X$	Commutativity of $\vee, \wedge$
L4. $X \vee (Y \vee Z) = (X \vee Y) \vee Z$	L4′. $X \wedge (Y \wedge Z) = (X \wedge Y) \wedge Z$	Associativity of $\vee, \wedge$
L5. $X \vee (X \wedge Y) = X$	L5′. $X \wedge (X \vee Y) = X$	Absorption between $\vee, \wedge$
L6. $X \leq Y \iff Y = X \vee Y$	L6′. $X \leq' Y \iff Y = X \wedge Y$	Consistency of $\vee, \wedge$ with partial order $\leq$
L7. $O \vee X = X$	L7′. $I \wedge X = X$	Identities of $\vee, \wedge$
L8. $I \vee X = I$	L8′. $O \wedge X = O$	Absorbing nulls of $\vee, \wedge$
L9. $X \vee (Y \wedge Z) = (X \vee Y) \wedge (X \vee Z)$	L9′. $X \wedge (Y \vee Z) = (X \wedge Y) \vee (X \wedge Z)$	Distributivity of $\vee, \wedge$
WL10. $a \star X \in \mathcal{W}$	WL10′. $a \star' X \in \mathcal{W}$	Closure of $\star, \star'$
WL11. $a \star (b \star X) = (a \star b) \star X$	WL11′. $a \star' (b \star' X) = (a \star' b) \star' X$	Associativity of $\star, \star'$
WL12. $a \star (X \vee Y) = a \star X \vee a \star Y$	WL12′. $a \star' (X \wedge Y) = a \star' X \wedge a \star' Y$	Distributive scalar–vector mult over vector sup/inf
WL13. $(a \vee b) \star X = a \star X \vee b \star X$	WL13′. $(a \wedge b) \star' X = a \star' X \wedge b \star' X$	Distributive scalar–vector mult over scalar sup/inf
WL14. $e \star X = X$	WL14′. $e' \star' X = X$	Scalar identities
WL15. $\perp \star X = O$	WL15′. $\top \star' X = I$	Scalar nulls
WL16. $a \star O = O$	WL16′. $a \star' I = I$	Vector nulls

all inf-$\star'$ combinations of elements in $\mathcal{A}$ is called its *inf-$\star'$ span*, denoted by $\mathrm{span}_\wedge(\mathcal{A})$. If $\mathcal{A} = \varnothing$, then $\mathrm{span}_\wedge(\mathcal{A}) = \{I\}$.

If the above sup-$\star$ and inf-$\star'$ combinations are based on a *finite* set of space elements, we shall call them *max-$\star$* and *min-$\star'$ combination*, respectively. A set $\mathcal{S}$ in a complete weighted lattice is called **max-$\star$ independent** if each point $F \in \mathcal{S}$ is not a max-$\star$ combination of points in $\mathcal{S} \setminus \{F\}$; otherwise, the set is called *max-$\star$ dependent*. Dually for the min-$\star'$ (in)dependence.

A max-$\star$ independent subset $\mathcal{B}$ of a CWL $\mathcal{W}$ is called an *upper basis* for the space if each space element F can be represented as a sup-$\star$ combination of basis elements:

$$F = \bigvee_i c_i \star B_i, \quad B_i \in \mathcal{B} \tag{28}$$

Dually, a min-$\star'$ independent subset $\mathcal{B}'$ of $\mathcal{W}$ is called a *lower basis* if $\mathcal{W} = \mathrm{span}_\wedge(\mathcal{B}')$. Examples of upper and lower bases are given later for CWLs of functions.

In this paper, we shall focus on CWLs whose underlying set is a *function space* $\mathcal{W} = \mathrm{Fun}(E, \mathcal{K})$ where E is an arbitrary nonempty set serving as the domain of our functions and the values of these functions are from a clodum $(\mathcal{K}, \vee, \wedge, \star, \star')$ of scalars as described in Sect. 3.1. Then, we extend *pointwise* the supremum, infimum and scalar multiplications of $\mathcal{K}$ to the functions: for $F, G \in \mathcal{W}$, $a \in \mathcal{K}$ and $x \in E$

$$\begin{aligned} (F \vee G)(x) &\triangleq F(x) \vee G(x) \\ (F \wedge G)(x) &\triangleq F(x) \wedge G(x) \\ (a \star F)(x) &\triangleq a \star F(x) \\ (a \star' F)(x) &\triangleq a \star' F(x) \end{aligned} \tag{29}$$

Under the first two operations $\mathcal{W}$ becomes a complete infinitely distributive lattice that inherits many properties from the lattice structure of $\mathcal{K}$. The least (O) and greatest (I) elements of $\mathcal{W}$ are the constant functions $O(x) = \perp$ and $I(x) = \top$, $x \in E$. Further, the scalar operations $\star$ and $\star'$, extended pointwise to functions, distribute over any suprema and infima, respectively. Thus, the function space $\text{Fun}(E, \mathcal{K})$ is by construction a *complete weighted lattice of functions* over the clodum $\mathcal{K}$. The collection of all its properties creates a rich algebraic structure.

If the clodum $\mathcal{K}$ is self-conjugate, then we can extend the conjugation $(\cdot)^*$ to functions F pointwise: $F^*(x) \triangleq (F(x))^*$. This obeys the same rules as the scalar conjugation on the clodum. Namely,

$$\left(\bigvee_i F_i\right)^* = \bigwedge_i {F_i}^*, \quad \left(\bigwedge_i G_i\right)^* = \bigvee_i {G_i}^*, \quad (a \star F)^* = a^* \star' F^* \tag{30}$$

In such a case, we have a *self-conjugate* complete weighted lattice.

The space of vectors and the space of signals with values from $\mathcal{K}$ are special cases of function lattices. In particular, if $E = \{1, 2, \ldots, n\}$, then $\mathcal{W}$ becomes the set of all n-dimensional vectors with elements from $\mathcal{K}$. If $E = \mathbb{Z}$, then $\mathcal{W}$ becomes the set of all discrete-time signals with values from $\mathcal{K}$.

Elementary increasing operators on $\mathcal{W}$ are those that act as **vertical translations** (in short V-translations) of functions. Specifically, pointwise 'multiplications' of functions $F \in \mathcal{W}$ by scalars $a \in \mathcal{K}$ yield the *V-translations* τ_a and *dual V-translations* τ'_a, defined by

$$[\tau_a(F)](x) \triangleq a \star F(x), \quad [\tau'_a(F)](x) \triangleq a \star' F(x) \tag{31}$$

A function operator ψ on $\mathcal{W}$ is called **V-translation invariant** if it commutes with any V-translation τ, i.e., $\psi\tau = \tau\psi$. Similarly, ψ is called *dual V-translation invariant* if $\psi\tau' = \tau'\psi$ for any dual V-translation τ'.

The above CWL $\mathcal{W}$ of functions contains an upper basis $\mathcal{B}$ and a lower basis $\mathcal{B}'$ which consist of the *impulse functions* q and the *dual impulses* q', respectively:

$$q_y(x) \triangleq \begin{cases} e, & x = y \\ \perp, & x \neq y \end{cases}, \quad q'_y(x) \triangleq \begin{cases} e', & x = y \\ \top, & x \neq y \end{cases} \tag{32}$$

Then, every function $F(x)$ admits a representation as a supremum of V-translated impulses placed at all points or as infimum of dual V-translated impulses:

$$F(x) = \bigvee_{y \in E} F(y) \star q_y(x) = \bigwedge_{y \in E} F(y) \star' q'_y(x) \tag{33}$$

By using the V-translations and the basis representation of functions with impulses, we can build more complex increasing operators, as explained next.

In general, if the space $\mathcal{W}$ is self-conjugate and has an upper basis $\mathcal{B}$, then it will also possess a lower basis since (28) implies that $F^* = \bigwedge_i {c_i}^* \star' {B_i}^*$. Thus, in the case of function CWLs that are self-conjugate, the upper and lower bases have the same cardinality, which is called the *dimension*[7] of $\mathcal{W}$. If this is finite, the space is called finite-dimensional; otherwise, it is called infinite-dimensional. Specific examples of finite- and infinite-dimensional upper and lower basis are mentioned in Sects. 3.3 and 3.4 for vector and signal spaces, respectively.

Consider systems that are V-translation-invariant dilations or erosions over $\mathcal{W}$. This invariance implies that they obey an interesting *nonlinear superposition principle* which has direct conceptual analogies with the well-known linear superposition. Specifically, we define δ to be a **dilation V-translation-invariant (DVI)** system iff for any $c_i \in \mathcal{K},\ \ F_i \in \mathcal{W}$

$$\delta\left(\bigvee_{i \in J} c_i \star F_i\right) = \bigvee_{i \in J} c_i \star \delta(F_i), \tag{34}$$

for any (finite or infinite) index set J. We also define ε to be an **erosion V-translation-invariant (EVI)** system iff

$$\varepsilon\left(\bigwedge_{i \in J} c_i \star' F_i\right) = \bigwedge_{i \in J} c_i \star' \varepsilon(F_i) \tag{35}$$

Compare the two above nonlinear superpositions with the *linear* superposition obeyed by a linear system Γ:

$$\Gamma\left(\sum_{i \in J} a_i \cdot F_i\right) = \sum_{i \in J} a_i \cdot \Gamma(F_i) \tag{36}$$

where J is a finite index set, a_i are constants from a field (of real or complex numbers) and F_i are field-valued signals from a linear space.

The structure of a DVI or EVI system's output is simplified if we express it via the system's impulse responses, defined next. Given a dilation system δ, its **impulse response map** is the map $H : E \to \text{Fun}(E, \mathcal{K})$ defined at each $y \in E$ as the output function $H(x, y)$ from δ when the input is the impulse $q_y(x)$. Dually, for an erosion operator ε we define its *dual impulse response map* H' via its outputs when excited by dual impulses: for $x, y \in E$

$$H(x, y) \triangleq \delta(q_y)(x), \quad H'(x, y) \triangleq \varepsilon(q'_y)(x) \tag{37}$$

[7] A dimension theory for semimodules has been developed in [65]. Further, the concept of an upper basis has been used in [15] to define the dimension of finite-dimensional subspaces of max-plus matrix algebra.

Applying a DVI operator δ or an EVI operator ε to (33) and using the definitions in (37) proves the following unified representation for all V-translation-invariant dilation or erosion systems.

Theorem 1 *(a) A system δ on $\mathcal{W}$ is DVI, i.e., obeys the sup-$\star$ superposition of* (34)*, if and only if its output can be expressed as*

$$\delta(F)(x) = \bigvee_{y \in E} H(x, y) \star F(y) \tag{38}$$

where H is its impulse response map in (37)*. (b) A system ε on $\mathcal{W}$ is EVI, i.e., obeys the inf-$\star'$ superposition of* (35)*, if and only if its output can be expressed as*

$$\varepsilon(F)(x) = \bigwedge_{y \in E} H'(x, y) \star' F(y) \tag{39}$$

where H' is its dual impulse response map in (37).

The result (38) for the max-plus dioid is analyzed in [2]. In the case of a signal space where $E = \mathbb{Z}$, the operations in (38) and (39) are like *time-varying nonlinear convolutions* where a dilation (resp. erosion) system's output is obtained as supremum (resp. infimum) of various impulse response signals produced by exciting the system with impulses at all points and weighted by the input signal values via a $\star$-'multiplication.'

3.3 Weighted lattice of vectors

Consider now the nonlinear vector space $\mathcal{W} = \mathcal{K}^n$, equipped with the pointwise partial ordering $\boldsymbol{x} \leq \boldsymbol{y}$, supremum $\boldsymbol{x} \vee \boldsymbol{y} = [x_i \vee y_i]$ and infimum $\boldsymbol{x} \wedge \boldsymbol{y} = [x_i \wedge y_i]$ between any vectors $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{W}$. Then, $(\mathcal{W}, \vee, \wedge, \star, \star')$ is a complete weighted lattice. Elementary increasing operators are the *vector V-translations* $\tau_a(\boldsymbol{x}) = a \star \boldsymbol{x} = [a \star x_i]$ and their duals $\tau'_a(\boldsymbol{x}) = a \star' \boldsymbol{x}$, which 'multiply' a scalar a with a vector $\boldsymbol{x}$ componentwise. A vector transformation on $\mathcal{W}$ is called (dual) V-translation invariant if it commutes with any vector (dual) V-translation.

By defining as 'impulse functions' the impulse vectors $\boldsymbol{q}_j = [q_j(i)]$ and their duals $\boldsymbol{q}'_j = [q'_j(i)]$, where the index j signifies the position of the identity, each vector $\boldsymbol{x} = [x_1, \ldots, x_n]^{\mathrm{T}}$ has a basis representation as a max of V-translated impulse vectors or as a min of V-translated dual impulse vectors:

$$\boldsymbol{x} = \bigvee_{j=1}^{n} x_j \star \boldsymbol{q}_j = \bigwedge_{j=1}^{n} x_j \star' \boldsymbol{q}'_j \tag{40}$$

More complex examples of increasing operators on this vector space are the max-$\star$ and the min-$\star'$ 'multiplications' of a matrix $\boldsymbol{M}$ with an input vector $\boldsymbol{x}$,

$$\delta_{\boldsymbol{M}}(\boldsymbol{x}) \triangleq \boldsymbol{M} \boxast \boldsymbol{x}, \quad \varepsilon_{\boldsymbol{M}}(\boldsymbol{x}) \triangleq \boldsymbol{M} \boxast' \boldsymbol{x} \tag{41}$$

which are, respectively, a vector dilation and a vector erosion. These two nonlinear matrix–vector ‘products’ are the prototypes of any vector transformation that obeys a sup-$\star$ or an inf-$\star'$ superposition, as proven next.

Theorem 2 *(a) Any vector transformation on the complete weighted lattice $\mathcal{W} = \mathcal{K}^n$ is DVI, i.e., obeys the sup-$\star$ superposition of (34), iff it can be represented as a matrix–vector max-$\star$ product $\delta_{\boldsymbol{M}}(\boldsymbol{x}) = \boldsymbol{M} \boxast \boldsymbol{x}$ where $\boldsymbol{M} = [m_{ij}]$ with $m_{ij} = \{\delta(\boldsymbol{q}_j)\}_i$.*

(b) Any vector transformation on $\mathcal{K}^n$ is EVI, i.e., obeys the inf-$\star'$ superposition of (35), iff it can be represented as a matrix–vector min-$\star'$ product $\varepsilon_{\boldsymbol{M}}(\boldsymbol{x}) = \boldsymbol{M} \boxast' \boldsymbol{x}$ where $\boldsymbol{M} = [m_{ij}]$ with $m_{ij} = \{\varepsilon(\boldsymbol{q}'_j)\}_i$.

Proof This is a special case of Theorem 1 where the domain points $x, y \in E$ become indices $i, j \in \{1, \ldots, n\}$ and the impulse response values $H(x, y)$ become matrix elements m_{ij}. Thus, the operations (38) and (39) become the max-$\star$ and min-$\star'$ products (41) of input vectors with the matrix $\boldsymbol{M} = [m_{ij}]$. □.

Given a vector dilation $\delta(\boldsymbol{x}) = \boldsymbol{M} \boxast \boldsymbol{x}$ with matrix $\boldsymbol{M} = [m_{ij}]$, there corresponds a unique adjoint vector erosion ε so that (δ, ε) is a *vector adjunction* on $\mathcal{W}$, i.e.,

$$\delta(\boldsymbol{x}) \leq \boldsymbol{y} \Longleftrightarrow \boldsymbol{x} \leq \varepsilon(\boldsymbol{y}) \tag{42}$$

(We seek adjunctions because they can easily generate projections.)

We can find the adjoint vector erosion by decomposing both vector operators based on scalar operators (η, ζ) that form a *scalar adjunction* on $\mathcal{K}$:

$$\eta(a, v) \leq w \Longleftrightarrow v \leq \zeta(a, w) \tag{43}$$

If we use as scalar ‘multiplication’ a commutative binary operation $\eta(a, v) = a \star v$ that is a dilation on $\mathcal{K}$, its scalar adjoint erosion becomes

$$\zeta(a, w) = \sup\{v \in \mathcal{K} : a \star v \leq w\} \tag{44}$$

which is a (possibly noncommutative) binary operation on $\mathcal{K}$. Then, the original vector dilation $\delta(\boldsymbol{x}) = \boldsymbol{M} \boxast \boldsymbol{x}$ is decomposed as

$$\{\delta(\boldsymbol{x})\}_i = \bigvee_j \eta(m_{ij}, x_j) = m_{ij} \star x_j, \quad i = 1, \ldots, n \tag{45}$$

whereas its adjoint vector erosion is decomposed as

$$\{\varepsilon(\boldsymbol{y})\}_j = \bigwedge_i \zeta(m_{ji}, y_i), \quad j = 1, \ldots, n \tag{46}$$

The latter can be written as a min-ζ matrix–vector multiplication

$$\varepsilon(\boldsymbol{y}) = \boldsymbol{M}^{\mathrm{T}} \Box'_{\zeta}\, \boldsymbol{y} \tag{47}$$

where the symbol $\Box'_\zeta$ denotes the following nonlinear product of a matrix $\boldsymbol{A} = [a_{ij}]$ with a matrix $\boldsymbol{B} = [b_{ij}]$:

$$\{\boldsymbol{A}\Box'_\zeta \boldsymbol{B}\}_{ij} \triangleq \bigwedge_k \zeta(a_{ik}, b_{kj})$$

Further, if $\mathcal{K}$ is a *clog*, it can be shown that $\zeta(a, w) = a^* \star' w$ and hence

$$\varepsilon(\boldsymbol{y}) = \boldsymbol{M}^* \boxed{\star}' \boldsymbol{y}, \tag{48}$$

where $\boldsymbol{M}^*$ is the *adjoint* (i.e., conjugate transpose)[8] of $\boldsymbol{M} = [m_{ij}]$:

$$\boldsymbol{M}^* \triangleq [m_{ji}{}^*] \tag{49}$$

Examples 4 (a) In the max-plus clog $(\overline{\mathbb{R}}, \vee, \wedge, +)$, consider the max-sum product (19) of a matrix $\boldsymbol{M}$ and a vector $\boldsymbol{x}$:

$$\boldsymbol{M} = \begin{bmatrix} 1 & 0.4 & 0 \\ 0.3 & 1 & 0.5 \\ 0.7 & 0.2 & 1 \end{bmatrix}, \quad \boldsymbol{x} = \begin{bmatrix} -0.2 \\ -0.6 \\ -0.3 \end{bmatrix} \Longrightarrow \delta_s(\boldsymbol{x}) = \boldsymbol{M} \boxplus \boldsymbol{x} = \begin{bmatrix} 0.8 \\ 0.4 \\ 0.7 \end{bmatrix} = \boldsymbol{y} \tag{50}$$

Let us apply to the result $\boldsymbol{y}$ the adjoint erosion. By (48) and (20),

$$\boldsymbol{M}^* = \begin{bmatrix} -1 & -0.3 & -0.7 \\ -0.4 & -1 & -0.2 \\ 0 & -0.5 & -1 \end{bmatrix} \Longrightarrow \varepsilon_s(\boldsymbol{y}) = \boldsymbol{M}^* \boxplus' \boldsymbol{y} = \boldsymbol{x} \tag{51}$$

Thus, in this example we have $\varepsilon_s \delta_s = \mathbf{id}$.

(b) In the clodum $([0, 1], \vee, \wedge, \min, \max)$, let us use a vector dilation δ_f as in (45) with max–min arithmetic (common in fuzzy systems), i.e., with $\eta(a, v) = a \star v = \min(a, v)$, to multiply the same matrix $\boldsymbol{M} = [m_{ij}]$ as above with a different vector $\boldsymbol{z}$ so as to reach the same result $\boldsymbol{y}$:

$$\boldsymbol{z} = \begin{bmatrix} 0.8 \\ 0.4 \\ 0.4 \end{bmatrix} \Longrightarrow \left[\delta_f(\boldsymbol{z})_i\right] = \left[\bigvee_j \min(m_{ij}, z_j)\right] = \boldsymbol{y} \tag{52}$$

To apply now the adjoint vector erosion (46), we need first to find the adjoint scalar erosion:

$$\zeta(a, w) = \sup\{v \in [0, 1] : \min(a, v) \leq w\} = \begin{cases} w, & w < a \\ 1, & w \geq a \end{cases} \tag{53}$$

[8] Despite its notation [15,21], $\boldsymbol{M}^*$ is not the elementwise conjugate of the matrix $\boldsymbol{M}$ but is obtained via transposition and elementwise conjugation of $\boldsymbol{M}$. To avoid the above ambiguity, we prefer the terminology 'adjoint' which is based on some conceptual similarities with the adjoint of a linear operator in Hilbert spaces [21].

Then, by (46) we can construct the adjoint vector erosion ε_f, from which we obtain $\varepsilon_f(\boldsymbol{y}) = \boldsymbol{z}$; i.e., again the adjoint vector erosion happened to be the inverse of the vector dilation.

Dually, given a vector erosion $\varepsilon'(\boldsymbol{y}) = \boldsymbol{M} \boxast' \boldsymbol{y}$ we can obtain its adjoint vector dilation δ' by starting from the 'dual multiplication' $\zeta(a, w) = a \star' w$ as a scalar erosion and finding its adjoint scalar dilation

$$\eta(a, v) = \inf\{w : a \star' w \geq v\} \tag{54}$$

Then the min-ζ matrix–vector multiplication $\varepsilon'(\boldsymbol{y}) = \boldsymbol{M} \boxast' \boldsymbol{y}$ with

$$\{\varepsilon'(\boldsymbol{y})\}_i = \bigwedge_j \zeta(m_{ij}, y_j) = m_{ij} \star' y_j, \quad i = 1, \ldots, n \tag{55}$$

has as adjoint a max-η matrix–vector multiplication $\delta'(\boldsymbol{x})$ with

$$\{\delta'(\boldsymbol{x})\}_j \triangleq \bigvee_i \eta(m_{ji}, x_i), \quad j = 1, \ldots, n \tag{56}$$

We can write this as a max-η matrix–vector multiplication

$$\delta'(\boldsymbol{x}) = \boldsymbol{M}^{\mathrm{T}} \Box_\eta \boldsymbol{x} \tag{57}$$

where the symbol $\Box_\eta$ denotes the following nonlinear product of a matrix $\boldsymbol{A} = [a_{ij}]$ with a matrix $\boldsymbol{B} = [b_{ij}]$:

$$\{\boldsymbol{A} \Box_\eta \boldsymbol{B}\}_{ij} \triangleq \bigvee_k \eta(a_{ik}, b_{kj})$$

Further, if $\mathcal{K}$ is a *clog*, it can be shown that $\eta(a, v) = a^* \star v$ and hence

$$\delta'(\boldsymbol{x}) = \boldsymbol{M}^* \boxast \boldsymbol{x} \tag{58}$$

3.4 Weighted lattice of signals

Consider the set $\mathcal{W} = \mathrm{Fun}(\mathbb{Z}, \mathcal{K})$ of all discrete-time signals $f : \mathbb{Z} \to \mathcal{K}$ with values from $\mathcal{K}$. Equipped with pointwise supremum $\vee$ and infimum $\wedge$, and two pointwise scalar multiplications ($\star$ and $\star'$), this becomes a complete weighted lattice $\mathcal{W}$ with partial order the pointwise signal relation $\leq$. The signal translations are the operators $\tau_{k,v}(f)(t) = f(t-k) \star v$, where $(k, v) \in \mathbb{Z} \times \mathcal{K}$ and $f(t)$ is an arbitrary input signal. Similarly, we define dual signal translations $\tau'_{k,v}(f)(t) = f(t-k) \star' v$. A signal operator on $\mathcal{W}$ is called *(dual) translation invariant* iff it commutes with any such (dual) translation. Note that, the above translation-invariance contains both a vertical translation and a horizontal translation; the horizontal part is the well-known *time-invariance*. Consider two elementary signals, called the *impulse* q and the *dual impulse* q':

$$q(t) \triangleq \begin{cases} e, & t = 0 \\ \perp, & t \neq 0 \end{cases}, \quad q'(t) \triangleq \begin{cases} e', & t = 0 \\ \top, & t \neq 0 \end{cases}$$

Then every signal f has a basis representation as a supremum of translated impulses or as infimum of dual translated impulses:

$$f(t) = \bigvee_k f(k) \star q(t-k) = \bigwedge_k f(k) \star' q'(t-k) \tag{59}$$

Consider now operators Δ on $\mathcal{W}$ that are dilations and translation invariant. Then Δ is both DVI in the sense of (34) and time invariant. We call such operators **dilation translation-invariant (DTI)** systems. Applying Δ to an input signal f decomposed as in (59) yields its output as the sup-$\star$ convolution $\circledast$ of the input with the system's impulse response $h = \Delta(q)$:

$$\Delta(f)(t) = (f \circledast h)(t) = \bigvee_{k \in \mathbb{Z}} f(k) \star h(t-k) \tag{60}$$

Conversely, every sup-$\star$ convolution is a DTI system. As done for the vector operators, we can also build signal operator pairs $(\Delta, \mathcal{E})$ that form adjunctions:

$$\Delta(f) \leq g \Longleftrightarrow f \leq \mathcal{E}(g) \tag{61}$$

Given Δ we can find its adjoint $\mathcal{E}$ from scalar adjunctions (η, ζ). Thus, by (43) and (44), if $\eta(h, f) = h \star f$, the adjoint signal erosion becomes

$$\mathcal{E}(g)(t) = \bigwedge_{\ell \in \mathbb{Z}} \zeta[h(\ell - t), g(\ell)] \tag{62}$$

Further, if $\mathcal{K}$ is a clog, then

$$\mathcal{E}(g)(t) = \bigwedge_{\ell \in \mathbb{Z}} g(\ell) \star' h^*(\ell - t) \tag{63}$$

Dually, if we start from an operator $\mathcal{E}$ on $\mathcal{W}$ that is erosion and translation invariant, then $\mathcal{E}$ is both EVI in the sense of (35) and time invariant. We call such operators **erosion translation-invariant (ETI)** systems. Applying $\mathcal{E}$ to an input signal g decomposed as in (59) yields the output as the inf-$\star'$ convolution $\circledast'$ of the input with the system's dual impulse response $h' = \mathcal{E}(q')$:

$$\mathcal{E}(g)(t) = (g \circledast' h')(t) = \bigwedge_{k \in \mathbb{Z}} g(k) \star' h'(t-k) \tag{64}$$

Setting $\zeta(h', g) = h' \star' g$ and using (43), (54) yields the adjoint signal dilation

$$\Delta(f)(t) = \bigvee_{\ell \in \mathbb{Z}} \eta[h'(\ell - t), f(\ell)] \tag{65}$$

which, if $\mathcal{K}$ is a clog, becomes

$$\Delta(f)(t) = \bigvee_{\ell \in \mathbb{Z}} f(\ell) \star h'^{*}(\ell - t) \tag{66}$$

An outcome of the previous discussion is:

Theorem 3 *(a) An operator Δ on a CWL $\mathcal{W}$ of signals is a dilation translation-invariant (DTI) system iff it can be represented as the sup-$\star$ convolution of the input signal with the system's impulse response $h = \Delta(q)$.*
(b) An operator $\mathcal{E}$ on $\mathcal{W}$ is an erosion dual-translation-invariant (ETI) system iff it can be represented as the inf-$\star'$ convolution of the input signal with the system's dual impulse response $h' = \mathcal{E}(q')$.

The above result for the max-plus clog was obtained in [44].

4 State and output responses

Based on the state-space model of a max-$\star$ dynamical system (2), we can compute its state response and output response if we know its *transition matrix*:

$$\boldsymbol{\Phi}(t_2, t_1) \triangleq \begin{cases} \boldsymbol{A}(t_2) \boxed{\star} \cdots \boxed{\star} \boldsymbol{A}(t_1 + 1) & \text{if} \quad t_2 > t_1 \\ \boldsymbol{I}_n & \text{if} \quad t_2 = t_1 \end{cases} \tag{67}$$

for $t_2 \geq t_1$, where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix in max-$\star$ matrix algebra that has values equal to the identity element e on its diagonal and least element (null) $\perp$ off-diagonally. The importance of $\boldsymbol{\Phi}$ is obvious by noticing that for a null input, the solution of the homogeneous state equation

$$\boldsymbol{x}(t) = \boldsymbol{A}(t) \boxed{\star} \boldsymbol{x}(t-1) \tag{68}$$

equals

$$\boldsymbol{x}(t) = \boldsymbol{\Phi}(t, 0) \boxed{\star} \boldsymbol{x}(0) \tag{69}$$

The transition matrix obeys a *semigroup property*:

$$\boldsymbol{\Phi}(t_2, t_1) \boxed{\star} \boldsymbol{\Phi}(t_1, t_0) = \boldsymbol{\Phi}(t_2, t_0), \quad t_0 \leq t_1 \leq t_2 \tag{70}$$

4.1 Time-varying systems

By using induction on (2), we can find the state and output responses of the general time-varying causal system; for $t = 0, 1, 2, \ldots,$

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{\Phi}(t,0) \boxast \boldsymbol{x}(0) \vee \left(\bigvee_{k=0}^{t} \boldsymbol{\Phi}(t,k) \boxast \boldsymbol{B}(k) \boxast \boldsymbol{u}(k) \right) \\ \boldsymbol{y}(t) &= \underbrace{\boldsymbol{C}(t) \boxast \boldsymbol{\Phi}(t,0) \boxast \boldsymbol{x}(0)}_{\boldsymbol{y}_{ni}(t) \triangleq \text{'null'-input resp.}} \vee \\ & \underbrace{\left(\bigvee_{k=0}^{t} \boldsymbol{C}(t) \boxast \boldsymbol{\Phi}(t,k) \boxast \boldsymbol{B}(k) \boxast \boldsymbol{u}(k) \right) \vee \boldsymbol{D}(t) \boxast \boldsymbol{u}(t)}_{\boldsymbol{y}_{ns}(t) \triangleq \text{'null'-state resp.}} \end{aligned} \tag{71}$$

where the supremum $\bigvee_{k=0}^{t}$ is null if $t < 0$. Henceforth, without loss of generality in (71), we shall assume that in practice $\boldsymbol{u}(0)$ is null (i.e., the input starts being active from $t \geq 1$) and use $\boldsymbol{x}(0)$ as the system's effective initial condition. (Otherwise, we use $\boldsymbol{x}(-1)$ as initial condition.) Thus, the output response is found to consist of two parts: (i) the 'null'-input response which is due only to the initial conditions $\boldsymbol{x}(0)$ and assumes a null input, i.e., equal to $\boldsymbol{u}(t) = \bot$, and (ii) the 'null'-state response which is due only to the input $\boldsymbol{u}(t)$ and assumes null initial conditions, i.e., $\boldsymbol{x}(0) = \bot$.

We observe that the 'null'-state response is essentially a time-varying sup-$\star$ matrix convolution

$$\boldsymbol{y}_{ns}(t) = \bigvee_{k=0}^{t} \boldsymbol{H}(t,k) \boxast \boldsymbol{u}(k) \tag{72}$$

of the input with a weight matrix

$$\boldsymbol{H}(t,k) \triangleq \boldsymbol{C}(t) \boxast \boldsymbol{\Phi}(t,k) \boxast \boldsymbol{B}(k) \vee q(t-k) \star \boldsymbol{D}(k)$$

The response (72) is a matrix version of the scalar time-varying sup-$\star$ convolution in (38).

The representation of the responses of time-varying max-$\star$ systems over idempotent dioids via the transition matrix has been developed in [40].

4.2 Time-invariant systems

Most of the results in this section are well known for time-invariant max-$\star$ systems over idempotent dioids, especially in the max-plus case [2]. We present them using monotone operators over weighted lattices.

Let the matrices $\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$ be constant. Then, the max-$\star$ state equations become:

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{A} \boxast \boldsymbol{x}(t-1) \ \vee \ \boldsymbol{B} \boxast \boldsymbol{u}(t) \\ \boldsymbol{y}(t) &= \boldsymbol{C} \boxast \boldsymbol{x}(t) \ \vee \ \boldsymbol{D} \boxast \boldsymbol{u}(t) \end{aligned} \tag{73}$$

Since the transition matrix simplifies to

$$\boldsymbol{\Phi}(t_2, t_1) = \boldsymbol{A}^{(t_2 - t_1)} \tag{74}$$

where $A^{(t)}$ denotes the t-fold max-$\star$ matrix product of $\boldsymbol{A}$ with itself for $t \geq 1$ and $\boldsymbol{A}^{(0)} = \boldsymbol{I}_n$, the solutions of the constant-matrix state equations become

$$\begin{aligned} \boldsymbol{x}(t) &= \boldsymbol{A}^{(t)} \boxed{\star} \boldsymbol{x}(0) \vee \left(\bigvee_{k=0}^{t} \boldsymbol{A}^{(t-k)} \boxed{\star} \boldsymbol{B} \boxed{\star} \boldsymbol{u}(k) \right) \\ \boldsymbol{y}(t) &= \underbrace{\boldsymbol{C} \boxed{\star} \boldsymbol{A}^{(t)} \boxed{\star} \boldsymbol{x}(0)}_{\boldsymbol{y}_{ni}(t) = \text{'null'-input resp.}} \\ &\quad \underbrace{\vee \boldsymbol{C} \boxed{\star} \left(\bigvee_{k=0}^{t} \boldsymbol{A}^{(t-k)} \boxed{\star} \boldsymbol{B} \boxed{\star} \boldsymbol{u}(k) \right) \vee \boldsymbol{D} \boxed{\star} \boldsymbol{u}(t)}_{\boldsymbol{y}_{ns}(t) = \text{'null'-state resp.}} \end{aligned} \tag{75}$$

By representing the matrix–vector $\star$-product as a dilation operator $\boldsymbol{x} \mapsto \delta_{\boldsymbol{A}}(\boldsymbol{x}) = \boldsymbol{A} \boxed{\star} \boldsymbol{x}$, we can express the state equations (73) with vector operators:

$$\begin{aligned} \boldsymbol{x}(t) &= \delta_{\boldsymbol{A}}[\boldsymbol{x}(t-1)] \ \vee \ \delta_{\boldsymbol{B}}[\boldsymbol{u}(t)] \\ \boldsymbol{y}(t) &= \delta_{\boldsymbol{C}}[\boldsymbol{x}(t)] \ \vee \ \delta_{\boldsymbol{D}}[\boldsymbol{u}(t)] \end{aligned} \tag{76}$$

and the state and output responses (75) in operator form:

$$\begin{aligned} \boldsymbol{x}(t) &= \delta_{\boldsymbol{A}}^{t}[\boldsymbol{x}(0)] \ \vee \ \left(\bigvee\nolimits_{k=0}^{t} \delta_{\boldsymbol{A}}^{t-k} \delta_{\boldsymbol{B}}[\boldsymbol{u}(k)] \right) \\ \boldsymbol{y}(t) &= \delta_{\boldsymbol{C}} \delta_{\boldsymbol{A}}^{t}[\boldsymbol{x}(0)] \vee \delta_{\boldsymbol{C}} \left(\bigvee_{k=0}^{t} \delta_{\boldsymbol{A}}^{t-k} \delta_{\boldsymbol{B}}[\boldsymbol{u}(k)] \right) \vee \delta_{\boldsymbol{D}}[\boldsymbol{u}(t)] \end{aligned} \tag{77}$$

For *single-input single-output (SISO)* systems, the mapping $u(t) \mapsto y_{ns}(t)$ can be viewed as a causal translation-invariant dilation system Δ. Hence, the 'null'-state response can be found as the sup-$\star$ convolution of the input with the system's impulse response $h = \Delta(q)$:

$$y_{ns}(t) = \Delta(u)(t) = (u \circledast h)(t) = \bigvee_{k=0}^{t} h(t-k) \star u(k) \tag{78}$$

The impulse response can be found from (75) by setting initial conditions equal to null and the input $u(t) = q(t)$:

$$h(t) = \begin{cases} \perp, & t < 0 \\ (\boldsymbol{C} \boxed{\star} \boldsymbol{B}) \vee D, & t = 0 \\ \boldsymbol{C} \boxed{\star} \boldsymbol{A}^{(t)} \boxed{\star} \boldsymbol{B}, & t > 0 \end{cases} \tag{79}$$

where in this case D is a scalar, $\boldsymbol{C}$ is a row vector and $\boldsymbol{B}$ a column vector. The last two results can be easily extended to multi-input multi-output (MIMO) systems by using an impulse response matrix as in (72).

5 Solving max-⋆ equations

Consider a scalar clodum $(\mathcal{K}, \vee, \wedge, \star, \star')$, a matrix $\boldsymbol{A} \in \mathcal{K}^{m\times n}$ and a vector $\boldsymbol{b} \in \mathcal{K}^m$. The set of solutions of the max-⋆ equation

$$\boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{b} \tag{80}$$

over $\mathcal{K}$ is either empty or forms a sup-semilattice. In [21] necessary and sufficient conditions are given for the existence and properties of such solutions in the max-plus case.

A related problem in applications of max-plus algebra to scheduling is when a vector $\boldsymbol{x}$ represents start times, a vector $\boldsymbol{b}$ represents finish times and the matrix $\boldsymbol{A}$ represents processing delays. Then, if $\boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{b}$ does not have an exact solution, it is possible to find the optimum $\boldsymbol{x}$ such that we minimize a norm of the earliness subject to zero lateness. We generalize this problem from max-plus to max-⋆ algebra. The optimum will be the solution of the following constrained minimization problem:

$$\text{Minimize } \|\boldsymbol{A} \boxed{\star} \boldsymbol{x} - \boldsymbol{b}\| \quad \text{s.t.} \quad \boldsymbol{A} \boxed{\star} \boldsymbol{x} \le \boldsymbol{b} \tag{81}$$

where the norm $||\cdot||$ is either the ℓ_∞ or the ℓ_1 norm. While the two above problems have been solved in [21] by using minimax algebra over the max-plus $(\overline{\mathbb{R}}, \vee, \wedge, +)$ or other clogs, we provide next an alternative and shorter proof of both results using adjunctions and for the general case when $\mathcal{K}$ may not be a clog.

Theorem 4 *Consider a vector dilation $\delta(\boldsymbol{x}) = \boldsymbol{A} \boxed{\star} \boldsymbol{x}$ over a scalar clodum $\mathcal{K}$ and let ε be its adjoint vector erosion. (a) If Eq.* (80) *has a solution, then*

$$\hat{\boldsymbol{x}} = \boldsymbol{A}^{\mathrm{T}} \Box'_{\zeta} \boldsymbol{b} = [\bigwedge_i \zeta(a_{ji}, b_i)] \tag{82}$$

is its greatest solution, where ζ is the scalar adjoint erosion (44) *of ⋆. (b) If $\mathcal{K}$ is a clog, the solution* (82) *becomes*

$$\hat{\boldsymbol{x}} = \boldsymbol{A}^{*} \boxed{\star}' \boldsymbol{b} \tag{83}$$

(c) The solution to the minimization problem (81) *is generally* (82)*, or* (83) *in the special case of a clog.*

Proof (a), (c): We showed in (46, 47) that the adjoint vector erosion of $\delta(\boldsymbol{x}) = \boldsymbol{A} \boxed{\star} \boldsymbol{x}$ is generally equal to $\varepsilon(\boldsymbol{y}) = \boldsymbol{A}^{\mathrm{T}} \Box'_{\zeta} \boldsymbol{y}$. Thus, the solution (82) has the form of an erosion, which by (16) has the property

$$\varepsilon(\boldsymbol{b}) = \bigvee \{\boldsymbol{x} : \delta(\boldsymbol{x}) \le \boldsymbol{b}\}$$

This implies that

$$\delta(\varepsilon(\boldsymbol{b})) = \bigvee \{\delta(\boldsymbol{x}) : \delta(\boldsymbol{x}) \le \boldsymbol{b}\}$$

The above immediately suggests that if $\varepsilon(\boldsymbol{b})$ is a solution, then it is the greatest solution. If not, then the difference $\boldsymbol{b} - \delta(\varepsilon(\boldsymbol{b}))$ is nonnegative and has the smallest ℓ_∞ or ℓ_1 norm. (b) For a clog, the scalar adjoint erosion of $\star$ is $\zeta(a, w) = a^* \star' w$ which gives (82) the simpler expression (83). □

A main idea behind the method for solving (81) is to consider vectors $\boldsymbol{x}$ that are *subsolutions* in the sense that $A \boxed{\star} \boldsymbol{x} \leq \boldsymbol{b}$ and find the greatest such subsolution. The set of subsolutions forms a sup-semilattice whose supremum equals $\hat{\boldsymbol{x}}$, which yields either the greatest exact solution of (80) or an optimum approximate solution in the sense of (81). Another attractive aspect of the adjunction-based solution is that it creates a lattice projection onto the max-$\star$ span of the columns of $\boldsymbol{A}$ via the opening $\delta(\varepsilon(\boldsymbol{b})) \leq \boldsymbol{b}$ that best approximates $\boldsymbol{b}$ from below.

Examples 5 (a) Consider solving $\delta_s(\boldsymbol{x}) = \boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{b}$ in the max-plus clog $(\overline{\mathbb{R}}, \vee, \wedge, +)$ with

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0.4 & 0 \\ 0.3 & 1 & 0.5 \\ 0.7 & 0.2 & 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} 0.8 \\ 0.4 \\ 0.9 \end{bmatrix} \tag{84}$$

The algorithm (83) yields the greatest solution

$$\hat{\boldsymbol{x}}_s = \varepsilon_s(\boldsymbol{b}) = \boldsymbol{A}^* \boxed{\star}' \boldsymbol{b} = [-0.2, -0.6, -0.1]^{\mathrm{T}} \tag{85}$$

among all exact solutions, which have the form $\boldsymbol{x} = [-0.2, c, -0.1]^{\mathrm{T}}$ with $c \leq -0.6$. Note that in Example 4(a) we had the same matrix but a different $\boldsymbol{b} = [0.8, 0.4, 0.7]^{\mathrm{T}}$ which gave a unique solution.

(b) Let us now try to solve $\delta_f(\boldsymbol{x}) = \boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{b}$ in the max–min clodum $([0, 1], \vee, \wedge, \min, \max)$ with the same $\boldsymbol{A}, \boldsymbol{b}$ as above. Then, by working as in Example 4(b) to construct the adjoint vector erosion, (82) yields

$$\hat{\boldsymbol{x}}_f = \varepsilon_f(\boldsymbol{b}) = \boldsymbol{A}^{\mathrm{T}} \Box'_\zeta \boldsymbol{b} = [0.8, 0.4, 0.4]^{\mathrm{T}} \tag{86}$$

where the specific ζ, i.e., the scalar adjoint erosion of $a \star v = \min(a, v)$, is given by (53). In this case, the algorithm gave an approximate solution since $\boldsymbol{A} \boxed{\star} \hat{\boldsymbol{x}}_f = [0.8, 0.4, 0.7]^{\mathrm{T}} \leq \boldsymbol{b}$. However, it is the greatest subsolution. Note that the same matrix but with a different $\boldsymbol{b}$ gave an exact solution in Example 4(b).

Further, by using adjunctions and duality, the CWL framework allows us to easily formulate and solve a *dual problem* of solving the min-$\star'$ equation

$$\boldsymbol{A} \boxed{\star}' \boldsymbol{y} = \boldsymbol{b} \tag{87}$$

either exactly if it has a solution, or by finding *supersolutions* $\boldsymbol{y}$ in the sense that $A \boxed{\star}' \boldsymbol{y} \geq \boldsymbol{b}$ and picking the smallest such supersolution. Approximate solutions of (87) can always be found by solving the following problem

$$\text{Minimize } \|\boldsymbol{A} \boxed{\star}' \boldsymbol{y} - \boldsymbol{b}\| \quad \text{s.t.} \quad \boldsymbol{A} \boxed{\star}' \boldsymbol{y} \geq \boldsymbol{b} \tag{88}$$

where the norm $||\cdot||$ is either the ℓ_∞ or the ℓ_1 norm. The set of supersolutions forms a semigroup under vector $\wedge$ whose infimum yields either the smallest exact solution of (87) if it exists or an optimum approximate solution in the sense of (88); this infimum is

$$\hat{\boldsymbol{y}} = \boldsymbol{A}^{\mathrm{T}} \Box_\eta \boldsymbol{b} \tag{89}$$

where η is the scalar adjoint dilation (54) of $\star'$. For a clog this becomes

$$\hat{\boldsymbol{y}} = \boldsymbol{A}^* \boxed{\star}\, \boldsymbol{b} \tag{90}$$

By viewing $\varepsilon(\boldsymbol{y}) = A \boxed{\star}' \boldsymbol{y}$ as a vector erosion, the operation in (89) or (90) is its corresponding adjoint vector dilation δ. This adjunction yields as best approximation the closing $\varepsilon(\delta(\boldsymbol{b})) \geq \boldsymbol{b}$ which is a lattice projection that comes optimally close to $\boldsymbol{b}$ from above.

Solving the one-sided equation (80) has direct applications in providing the system reachability and observability problems with exact or approximate solutions, as shown in Sect. 8. There are also double-sided max-$\star$ equations of the type

$$\boldsymbol{A} \boxed{\star}\, \boldsymbol{x} = \boldsymbol{B} \boxed{\star}\, \boldsymbol{y} \tag{91}$$

which model synchronization problems and can be solved by iterating the method (83) between left and right side, as shown in [22]. This has been extended in [29,42] to one- and two-sided equations whose matrix elements are intervals representing numerical uncertainties.

6 Spectral analysis in max-$\star$ algebra

There has been significant progress on eigenvalue analysis for the max-plus semiring $(\mathbb{R} \cup \{-\infty\}, \vee, +)$; see [15,21] and the references therein. Herein, we extend some of the main results to any scalar clodum[9] $\mathcal{K}$ even in cases where the 'multiplications' do not have inverses. The only constraint on the clodum $\mathcal{K}$ is to be *radicable* w.r.t. operations $\star, \star'$: namely, for each $a \in \mathcal{K}$ and integer $p \geq 2$ there is some $x \in \mathcal{K}$ such that its p-fold $\star$-multiplication with itself equals a, i.e., $x^{\star p} \triangleq x \star x \star \cdots \star x = a$. Note that both the max-plus clog and the max–min clodum are radicable.

Consider a $n \times n$ matrix $\boldsymbol{A} = [a_{ij}]$, $n > 1$. This can be represented by a *directed weighted graph* $\mathrm{Gr}(\boldsymbol{A})$ that has n nodes and arcs connecting pairs of nodes (i, j) if the corresponding weights $a_{ij} > \bot$. If $\mathrm{Gr}(\boldsymbol{A})$ is strongly connected, i.e., if there is a path from every node to every other node, then $\boldsymbol{A}$ is called *irreducible*. Consider a *path* on the graph, i.e., a sequence of nodes $\boldsymbol{\pi} = (i_0, i_1, \ldots, i_t)$ with length $\ell(\boldsymbol{\pi}) = t$; its weight is defined by $w(\boldsymbol{\pi}) \triangleq a_{i_0 i_1} \star \ldots \star a_{i_{t-1} i_t}$. A path $\boldsymbol{\sigma}$ is called a *cycle* if

[9] Although the main results [15] of max-plus eigenvalue analysis in the max-plus semiring assume all scalars $< +\infty$, in the more general max-$\star$ eigenvalue analysis over a clodum we allow scalars to equal $\top$; this has direct applications for cloda $\mathcal{K} = [0, 1]$ in fuzzy systems, like the max–min clodum, where $1 = e = \top$.

$i_0 = i_t$; the cycle is *elementary* if the nodes $i_0, \ldots, i_{t-1}$ are distinct. For any cycle $\boldsymbol{\sigma}$, we define its *cycle mean*[10] by $w(\boldsymbol{\sigma})^{\star(1/\ell(\boldsymbol{\sigma}))}$. Let

$$\lambda(\boldsymbol{A}) \triangleq \bigvee_{\text{all cycles } \boldsymbol{\sigma} \text{ of } \boldsymbol{A}} w(\boldsymbol{\sigma})^{\star(1/\ell(\boldsymbol{\sigma}))} \tag{92}$$

be the *maximum cycle mean* in $\mathrm{Gr}(\boldsymbol{A})$. Since $\mathrm{Gr}(\boldsymbol{A})$ has n nodes, only elementary cycles with length $\leq n$ need be considered in (92). There is also at least one cycle whose average weight coincides with the maximum value (92); such a cycle is called *critical*. The existence of $\lambda(\boldsymbol{A})$ is guaranteed if $(\mathcal{K}, \star)$ is radicable.

The max-$\star$ eigenproblem for the matrix $\boldsymbol{A}$ consists of finding its *eigenvalues* λ and *eigenvectors* $\boldsymbol{v} \neq \bot$ such that

$$\boldsymbol{A} \boxast \boldsymbol{v} = \lambda \star \boldsymbol{v} \tag{93}$$

The maximum cycle mean $\lambda(\boldsymbol{A})$ plays a fundamental role in this eigenproblem for many reasons [15,21]: It is the largest eigenvalue of $\boldsymbol{A}$ and the only eigenvalue whose corresponding eigenvectors may be finite. Thus, $\lambda(\boldsymbol{A})$ is called the *principal eigenvalue* of $\boldsymbol{A}$. Some further properties include the following. Define the *metric matrix* generated by $\boldsymbol{A}$ as the series

$$\boldsymbol{\Gamma}(\boldsymbol{A}) \triangleq \bigvee_{k=1}^{\infty} \boldsymbol{A}^{(k)} \tag{94}$$

If it converges, it conveys very useful information since its elements equal the weights of the heaviest paths of any length for all pairs of nodes (like a graph of longest distances), and its columns can provide eigenvectors [15,21]. However, its existence is controlled by $\lambda(\boldsymbol{A})$ as explained by

Theorem 5 *Assume a $n \times n$ matrix $\boldsymbol{A} = [a_{ij}]$ with elements from a radicable clodum $\mathcal{K}$. (a) The infinite series (94) converges in finite time to a matrix $\boldsymbol{\Gamma}(\boldsymbol{A}) = [\gamma_{ij}]$ if $\lambda(\boldsymbol{A}) \leq e$, in which case for all $t \geq 1$*

$$\boldsymbol{A}^{(t)} \leq \boldsymbol{\Gamma}(\boldsymbol{A}) = \boldsymbol{A} \vee \boldsymbol{A}^{(2)} \vee \cdots \vee \boldsymbol{A}^{(n)} \tag{95}$$

(b) If all $a_{ij} < \top$, then both (95) holds and all $\gamma_{ij} < \top$ if and only if $\lambda(\boldsymbol{A}) \leq e$.
(c) If $\lambda(\boldsymbol{A}) \leq e$ and $\boldsymbol{A}$ is irreducible, then all $\gamma_{ij} > \bot$.

Proof We extend the results of [15,21] to a general radicable clodum. (a) If $\lambda(\boldsymbol{A}) \leq e$, then a path $\boldsymbol{\pi}$ between any nodes i, j of length $> n$ contains cycles, all whose weights $\leq e$. By deleting these cycles, we can create only heavier subpaths $\boldsymbol{\pi}'$ of length $\leq n$, i.e., $w(\boldsymbol{\pi}') \geq w(\boldsymbol{\pi})$. Given the finite only number of paths (without cycles) between

[10] For the max-plus clog $(\overline{\mathbb{R}}, \vee, \wedge, +)$ the mean of a cycle $\boldsymbol{\sigma}$ is given by $w(\boldsymbol{\sigma})/\ell(\boldsymbol{\sigma})$, for the max-product clog $([0, \infty], \vee, \wedge, \times)$ it is given by $w(\boldsymbol{\sigma})^{1/\ell(\boldsymbol{\sigma})}$, whereas for the max–min clodum $([0, 1], \vee, \wedge, \min, \max)$ the cycle mean is simply $w\boldsymbol{\sigma}$.

any nodes i, j, if a path exists, then a heaviest such path also exists with length $\leq n$ and weight γ_{ij}; if no path exists, then $\gamma_{ij} = \perp$. (b) If $\lambda(\boldsymbol{A}) \leq e$, then in part (a) we proved convergence in finite time. Further, since all elements of $\boldsymbol{A}$ are $< \top$, the finite-length heaviest path between any nodes i, j will have weight $\gamma_{ij} < \top$. If $\lambda(\boldsymbol{A}) > e$, then there exists a cycle with weight $> e$ which will drive at least one element in $\boldsymbol{A}^{(t)}$ unbounded (i.e., $\top$) as $t \to \infty$ and hence there is no finite convergence. Thus, (95) holds iff $\lambda(\boldsymbol{A}) \leq e$. (c) If $\boldsymbol{A}$ is also irreducible, i.e., $\mathrm{Gr}(\boldsymbol{A})$ is strongly connected, then a path exists between any nodes i, j and hence each $\gamma_{ij} > \perp$. The above results also cover the case of cloda with $e = \top$ (like the max–min clodum) because then the condition $\lambda(\boldsymbol{A}) \leq e$ always holds. □

By using duality between the max-$\star$ and min-$\star'$ matrix subalgebras over a radicable scalar clodum we can also solve the *dual eigenproblem*

$$\boldsymbol{A} \boxdot' \boldsymbol{v}' = \lambda' \star' \boldsymbol{v}' \tag{96}$$

The *dual principal eigenvalue*, denoted by $\lambda'(\boldsymbol{A})$, is the smallest of all dual eigenvalues and can be found as the *minimum cycle mean* of $\boldsymbol{A}$.

7 Causality, stability

Assume for brievity SISO systems. (Our results can be easily extended for MIMO systems.) Assume also that systems' matrices are constant. A useful bound for signals $f(t)$ processed by max-$\star$ systems is their supremal value $\bigvee_t f(t)$. We call max-$\star$ systems *upper-stable* if an upper bounded input and initial condition yields an upper bounded output, i.e., if

$$\boldsymbol{x}(0) < \top \text{ and } \bigvee_t u(t) < \top \Longrightarrow \bigvee_t y(t) < \top \tag{97}$$

If initial conditions are null and (97) is satisfied, we call the system bounded-input bounded-output (BIBO) upper-stable. Dually, min-$\star'$ systems are called *lower-stable* if a lower bounded input and initial condition yields a lower bounded output, i.e., if

$$\boldsymbol{x}(0) > \perp \text{ and } \bigwedge_t u(t) > \perp \Longrightarrow \bigwedge_t y(t) > \perp \tag{98}$$

A max-$\star$ (min-$\star'$) dynamical system with null initial conditions can be viewed as a DTI (ETI) system mapping the input u to the output which is the sup-$\star$ (inf-$\star'$) convolution $y = u \circledast h$ ($y = u \circledast' h'$) of the input with the (dual) impulse response h (h'). The following theorem provides us with simple algebraic criteria for checking the causality and stability of DTI and ETI systems based on their impulse response.

Theorem 6 *(a1) Consider a DTI system Δ and let $h = \Delta(q)$ be its impulse response. Then: (a1) The system is causal iff $h(t) = \perp$ for all $t < 0$. (a2) The system is BIBO upper-stable iff $\bigvee_t h(t) < \top$.*

(b) Consider an ETI system $\mathcal{E}$ and let $h' = \mathcal{E}(q')$ be its dual impulse response. Then: (b1) The system is causal iff $h'(t) = \top$ for all $t < 0$. (b2) The system is BIBO lower-stable iff $\bigwedge_t h'(t) > \bot$.

Proof Part (a): (a1) follows from the definition of causality since the output can be written as $\Delta(u)(t) = \bigvee_k u(t-k) \star h(k)$. (a2) Sufficiency: If u and h have suprema $< \top$, then their dilation $y = u \circledast h$ also has a supremum $< \top$ because

$$y(t) \leq \bigvee_k u(k) \star \bigvee_k h(k), \quad \forall t$$

Necessity: Assume now that Δ is upper-stable. Then $\bigvee_t h(t)$ must be $< \top$, because otherwise we can find a bounded input yielding an unbounded output. For example, the input $u(t) = q(t)$ yields as output $y(t) = h(t)$. Obviously, this u is bounded, but if $\bigvee_t h(t) = \top$ we get an unbounded output. Part (b) follows by duality. □

The stability of a linear dynamical system can be expressed via the eigenvalues of its state transition matrix $\boldsymbol{A}$. For max-$\star$ (min-$\star'$) systems we derive below a conceptually similar result that links the upper (lower) stability of the system with the (dual) principal eigenvalue of $\boldsymbol{A}$.

Theorem 7 *(a) Consider a max-$\star$ system whose matrices do not contain any $\top$ elements. If $\lambda(\boldsymbol{A}) \leq e$, the system is upper-stable. (b) If a min-$\star'$ system has matrices without any $\bot$ elements and $\lambda'(\boldsymbol{A}) \geq e'$, then the system is lower-stable.*

Proof (a) By (79), if $\boldsymbol{C} = [c_i]^{\mathrm{T}}$ and $\boldsymbol{B} = [b_j]$,

$$h(t) = \max_i \max_j c_i \star a_{ij}^{(t)} \star b_j \tag{99}$$

where $a_{ij}^{(t)}$ is the (i, j) element of matrix $\boldsymbol{A}^{(t)}$. By Theorem 5, we have $\boldsymbol{A}^{(t)} \leq \boldsymbol{\Gamma}(\boldsymbol{A}) = [\gamma_{ij}]$, and equivalently $a_{ij}^{(t)} \leq \gamma_{ij} < \top$ for all $i, j, t \geq 1$. Thus,

$$\bigvee_t h(t) \leq \max_{i,j} \gamma_{ij} \star \max_i c_i \star \max_j b_j < \top \tag{100}$$

Hence, by Theorem 6 the system is BIBO upper-stable. This upper bounds the null-state response $y_{ns}(t)$ of the output. Now if $\boldsymbol{x}(0) \neq \bot$, the null-input response $y_{ni}(t) = \boldsymbol{C} \boxast \boldsymbol{A}^{(t)} \boxast \boldsymbol{x}(0)$ will also be upper bounded via a similar proof as above. Thus, the system is upper-stable. Part (b) follows by duality. □

From another viewpoint, the useful information in a signal f analyzed by a DTI system exists only at times where $f(t)$ is not null. Thus, its *support* (or effective domain) is defined by $\mathrm{Spt}_\vee(f) \triangleq \{t : f(t) > \bot\}$. An alternative useful bound for signals $f(t)$ processed by such systems is their supremal 'absolute value' over their support:

$$M_f \triangleq \bigvee_{t \in \mathrm{Spt}_\vee(f)} \mu(f(t)) \tag{101}$$

where $\mu(a) \triangleq a \vee a^*$ is called the *absolute value seminorm* in [21] and is 'sublinear' over a self-conjugate clodum in the sense that $\mu(a \vee b) \leq \mu(a) \vee \mu(b)$. We call max-$\star$ systems BIBO *absolutely stable* iff a bounded input yields a bounded output in the following sense:

$$M_u < \top \Longrightarrow M_y < \top \tag{102}$$

This is controlled by the system's impulse response as shown next.

Theorem 8 *Consider a DTI system Δ over a self-conjugate clodum whose matrices do not contain any $\top$ elements. Let $h = \Delta(q)$ be its impulse response. Then, the system is BIBO absolutely stable iff $M_h < \top$.*

Proof Sufficiency: If u and h have finite bounds M_u and M_h within their supports U and H, respectively, then their sup-$\star$ convolution $y = u \circledast h$ is also absolutely bounded because

$$\mu(y(t)) \leq \bigvee_{k \in U \cap (H^s)_{+t}} \mu[u(k) \star h(t-k)] \leq M_u \star M_h$$

for all t in the Minkowski set addition $U \oplus H = \{k + \ell : k \in U, \ell \in H\}$ of the two supports, where $(H^s)_{+t} = \{t - k : k \in H\}$ denotes the reflected H translated by t. Necessity: Assume that Δ is stable. Then M_h must be finite, because otherwise we can find a bounded input yielding an unbounded output. For example, the bounded input $u(t) = q(t)$ yields the output $y(t) = h(t)$ which is unbounded if $M_h = \top$. □

The next theorem links absolute stability with the principal eigenvalue of the system.

Theorem 9 *Consider a max-$\star$ system over a clog whose matrices do not contain any $\top$ elements. For matrix $\boldsymbol{A} = [a_{ij}]$ assume that it is irreducible, $a_{ii} > \perp$ for some i, and there is a unique critical cycle of length d corresponding to its finite principal eigenvalue $\lambda(\boldsymbol{A})$. Then: (a) If $\lambda(\boldsymbol{A}) = e$, the impulse response of the system is eventually periodic with period d. (b) The system is BIBO absolutely stable iff $\lambda(\boldsymbol{A}) = e$.*

Proof (a) As shown for the max-plus case in [18] under the above hypotheses for $\boldsymbol{A}$, if $\lambda(\boldsymbol{A}) = 0$, then $\boldsymbol{A}$ is order-d-periodical, i.e., there is an integer k_0 such that $\boldsymbol{A}^{(k+d)} = \boldsymbol{A}^{(k)}$ $\forall k \geq k_0$. The proof of the above in [18] can be extended to general clogs. Hence, by (79), there exists k_0 such that $h(k+d) = h(k)$ for all $k \geq k_0$. (b) Let $\lambda = \lambda(\boldsymbol{A})$. Then $\lambda^* \star \boldsymbol{A}$ is order-d-periodical and hence $\boldsymbol{A}^{(k+d)} = \lambda^{\star d} \star \boldsymbol{A}^{(k)}$ for all $k \geq k_0$. Hence,

$$h(k+d) = \lambda^{\star d} \star h(k), \quad \forall k \geq k_0 \tag{103}$$

Further, the absence of $\top$ values in the system's matrices guarantees that $h(k)$ does not have any such values. Now, if $\lambda = e$, then $h(k+d) = h(k)$ $\forall k \geq k_0$ and hence $M_h < \top$. In contrast, if $\lambda \neq e$, then (103) will drive asymptotically (as $k \to \infty$) the values of $\mu(h(k))$ unbounded, and hence $M_h = \top$. □

8 Reachability, observability

Assume single-input single-output systems with constant matrices described by (75), acting on a CWL over a clodum $\mathcal{K}$. A max-$\star$ system is called *reachable* in k-steps if the following system of nonlinear equations can be solved and provide the control input sequence $\boldsymbol{u}_k = [u(1), \ldots, u(k)]^{\mathrm{T}}$ required to drive the system from the initial state $\boldsymbol{x}(0)$ to any desired state $\boldsymbol{x}(k)$ in k steps:

$$\boldsymbol{x}(k) = \boldsymbol{A}^{(k)} \boxed{\star}\, \boldsymbol{x}(0) \ \vee \mathcal{C}_k \boxed{\star}\, \boldsymbol{u}_k \tag{104}$$

where $\mathcal{C}_k = [\boldsymbol{A}^{(k-1)} \boxed{\star}\, \boldsymbol{B}, \ldots, \boldsymbol{A} \boxed{\star}\, \boldsymbol{B}, \boldsymbol{B}]$ is called the *controllability matrix*. This system of max-$\star$ equations can be solved using the methods of Sect. 5. However, we can simplify it first by assuming that the input is dominating the initial conditions (e.g., by assuming inputs with sufficiently large values); then, the second term is greater than the first term of the right hand side, and we can rewrite (104) as

$$\mathcal{C}_k \boxed{\star}\, \boldsymbol{u}_k = \boldsymbol{x}(k) \tag{105}$$

If there is an exact solution to (105), the system is called *weakly reachable* [27]. Because of some dimensional anomalies in minimax algebra [21], there is no guarantee of exact solution even when $\mathcal{C}_k$ has adequate column rank[11] (i.e., n max-$\star$ independent columns) because the max-$\star$ span of its columns may be only a subset of $\mathcal{K}^n$, unlike the linear system case where full rank of $\mathcal{C}_k$ makes the system reachable in at most $k = n$ steps. Another difference with linear systems is that the max-$\star$ column rank may not be the same with the row rank. Thus, by using $k > n$ one may obtain a matrix $\mathcal{C}_k$ that will give an exact solution. By Theorem 4, if there exists an exact solution, the greatest solution is the lattice erosion

$$\hat{\boldsymbol{u}}_k = \varepsilon(\boldsymbol{x}(k)) = \mathcal{C}_k^{\mathrm{T}} \Box'_{\zeta} \boldsymbol{x}(k) \tag{106}$$

where ε is the adjoint erosion of the dilation $\delta(\boldsymbol{y}) = \mathcal{C}_k \boxed{\star}\, \boldsymbol{y}$. (See Sec.3.3.)

If $\mathcal{K}$ is a clog, the solution (106) becomes

$$\hat{\boldsymbol{u}}_k = \mathcal{C}_k^* \boxed{\star}'\, \boldsymbol{x}(k) \tag{107}$$

However, in certain applications Eq. (105) may be too strong of a condition and it may be sufficient to solve an approximate reachability problem that has some optimality aspects. Specifically, consider finding an optimal control input sequence $\boldsymbol{u}_k$ as solution to the following constrained optimization problem:

$$\text{Min} \quad \|\mathcal{C}_k \boxed{\star}\, \boldsymbol{u}_k - \boldsymbol{x}(k)\| \quad \text{s.t.} \quad \mathcal{C}_k \boxed{\star}\, \boldsymbol{u}_k \leq \boldsymbol{x}(k) \tag{108}$$

[11] The column (row) rank of a matrix over a clodum can be defined as the largest number of max-$\star$ independent columns (rows). In [15,21,28] there are also weaker concepts of vector independence in minimax algebra.

where the norm $\|\cdot\|$ is either the ℓ_∞ or the ℓ_1 norm. Then the optimal solution is (106) or (107).

Examples 6 Consider a max-sum system over the max-plus clog with

$$\boldsymbol{A} = \begin{bmatrix} -4 & -1 & -3 \\ -4 & -3 & 0 \\ 1 & -2 & -1 \end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix} -1 \\ 2 \\ -1 \end{bmatrix} \tag{109}$$

The controllability matrix for $k = 5$ steps (shown below) has full column rank (5 and larger than the row rank):

$$\mathcal{C}_5 = \begin{bmatrix} -1 & 1 & -2 & -1 & 1 \\ 2 & -1 & 0 & 2 & 1 \\ -1 & 0 & 2 & 1 & 0 \end{bmatrix} \tag{110}$$

(a) If $\boldsymbol{x}(5) = [1, 1, 1]^{\mathrm{T}}$ is the desired state, then this vector belongs to the max-plus span of the columns of $\mathcal{C}_5$ since

$$\mathcal{C}_5 \boxplus \begin{bmatrix} -1 \\ 0 \\ -1 \\ -1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix} \tag{111}$$

Thus, $\hat{\boldsymbol{u}} = [-1, 0, -1, -1, 0]^{\mathrm{T}}$ is the greatest solution among all possible 5-step control sequences that can reach the same state, which have values $[a, b, -1, d, 0]^{\mathrm{T}}$ with $a \le -1,\ b \le 0,\ d \le -1$.
(b) However, if the desired state is $\boldsymbol{x}(5) = [-3, 3, 0]^{\mathrm{T}}$ then this vector does not belong to the column span of $\mathcal{C}_5$. Indeed, (107) yields $\hat{\boldsymbol{u}} = [-2, -4, -2, -2, -4]^{\mathrm{T}}$ which is only a greatest subsolution of (105) since it can only reach $[-3, 0, 0]^{\mathrm{T}}$ which is a lower state than desired.

The above ideas can also be applied to the observability problem. A max-$\star$ system is *observable* if we can estimate the initial state by observing a sequence of output values. By (75), this can be done if the following system of nonlinear equations can be solved:

$$\begin{bmatrix} y(1) \\ \vdots \\ y(k) \end{bmatrix} = \underbrace{\begin{bmatrix} \boldsymbol{C} \boxed{\star} \boldsymbol{A} \\ \vdots \\ \boldsymbol{C} \boxed{\star} \boldsymbol{A}^{(k)} \end{bmatrix}}_{\mathcal{O}_k} \boxed{\star}\, \boldsymbol{x}(0) \ \vee \begin{bmatrix} y_{ns}(1) \\ \vdots \\ y_{ns}(k) \end{bmatrix} \tag{112}$$

Assuming that the first term of the right hand side containing the initial state dominates the second term that contains the input (e.g., by assuming inputs with sufficiently small values), we can rewrite the above as

$$\mathcal{O}_k \boxast \boldsymbol{x}(0) = \boldsymbol{y}_k = [y(1), \ldots, y(k)]^{\mathrm{T}} \tag{113}$$

This equation can be solved exactly or approximately by using the same methods as for the reachability equation. Thus, if $\mathcal{K}$ is a clog, the general solution is

$$\hat{\boldsymbol{x}}(0) = \mathcal{O}_k{}^* \boxast' \boldsymbol{y}_k \tag{114}$$

and has the property that it is the largest solution with $\mathcal{O}_k \boxast \hat{\boldsymbol{x}}(0) \leq \boldsymbol{y}_k$.

9 Applications, special cases

9.1 Max-sum systems

One broad class of nonlinear dynamical systems is described by (2) or (6) by using the max-plus clog $(\overline{\mathbb{R}}, \vee, \wedge, +)$ for scalar arithmetic and the max-sum $\boxplus$ and min-sum $\boxplus'$ matrix products (19),(20), which are the basis of minimax algebra [21].

Special cases of max-sum or min-sum dynamical systems have been used for modeling, control and optimization in (i) discrete event dynamical systems (DES) for applications including scheduling, manufacturing and transportation, (ii) shortest path and related dynamic programming problems, and (iii) operations research; see [2,15,16,18,19,23,27,30,37,38] and the references therein.

Next, we examine state-space formulations and stability issues for two classes of max-sum or min-sum dynamical systems modeling recursive nonlinear filtering and shortest path computation, which can be described by generalized versions of the max-sum recursion (8) or its dual.

9.1.1 State-space models of recursive nonlinear filters

A very large class of discrete linear time-invariant systems used in control and signal processing [14,53] is modeled via the following class of linear difference equations:

$$y(t) = \sum_{i=1}^{n} a_i y(t-i) + \sum_{j=0}^{m} b_j u(t-j) \tag{115}$$

Replacing sum with maximum and multiplication with addition gives us the following nonlinear *max-sum difference equation* [44]

$$y(t) = \left(\bigvee_{i=1}^{n} a_i + y(t-i) \right) \vee \left(\bigvee_{j=0}^{m} b_j + u(t-j) \right) \tag{116}$$

The signal values and all coefficients a_i, b_j are from the max-plus clog. If some $a_i = -\infty$, the term with $y(t-i)$ is not used in the equation. Special (mainly nonrecursive) cases of such nonlinear difference equations have found many applications in morphological signal and image processing [31,48,59,60], convex analysis [43,58], and optimization [3,4].

The max-plus version of the general state equations (2) can model the dynamics of recursive discrete-time filters described by the above max-sum difference equation. Specifically, if $m = 0$, setting $x_i(t) = y(t - n + i - 1)$, $i = 1, \ldots, n$, and choosing matrices

$$\begin{aligned} \boldsymbol{A} &= \begin{bmatrix} -\infty & 0 & -\infty & \ldots & -\infty \\ -\infty & -\infty & 0 & \ldots & -\infty \\ \vdots & \vdots & & & \vdots \\ -\infty & -\infty & -\infty & \ldots & 0 \\ a_n & a_{n-1} & a_{n-2} & \ldots & a_1 \end{bmatrix}, \quad B = [b_0] \\ \boldsymbol{C} &= [a_n, \ldots, a_1], \quad D = [b_0] \end{aligned} \tag{117}$$

models (116) as a max-sum special case of (2).

Consider now the following *min-sum difference equation*, which describes a dual system to that of (116):

$$y(t) = \left(\bigwedge_{i=1}^{n} a_i + y(t-i) \right) \wedge \left(\bigwedge_{j=0}^{m} b_j + u(t-j) \right) \tag{118}$$

Its dynamics can be modeled by the min-sum version of the general state equations (6). For $m = 0$, it admits a state-space model as in (117), the only difference being that the null elements in the system matrices should be $+\infty$.

The system described by (116) or (117) is a dilation time-invariant (DTI) system iff all its initial conditions are null ($-\infty$) and is initially at rest, i.e., if $u(t) = -\infty$ for $t \leq t_0$ then $y(t) = -\infty$ for $t \leq t_0$. Similar conditions apply for (118) to make it correspond to an erosion time-invariant (ETI) system.

Theorem 10 *The max-plus principal eigenvalue of the matrix* $\boldsymbol{A}$ *in* (117) *is equal to* $\lambda(\boldsymbol{A}) = \bigvee_{k=1}^{n} a_k/k$.

Proof The directed weighted graph of $\boldsymbol{A}$ has n nodes and n elementary cycles $(j, j+1, \ldots, n, j)$ for $j = 1, \ldots, n$, each with average weight $a_{n-j+1}/(n-j+1)$. Hence, $\lambda(\boldsymbol{A}) = \bigvee_{k=1}^{n} a_k/k$. □

Thus, the max-sum system corresponding to the recursive nonlinear filter described by (116) is upper-stable iff all the coefficients a_k are nonpositive and absolutely stable if additionally at least one of them is zero. Such a numerical example is shown in Fig. 2a, where Theorem 9 also applies and predicts a periodic impulse response. Further, responses from stable and unstable DTI and ETI systems are shown in Fig. 2. The stable outputs of Fig. 2c, d illustrate the applicability of recursive DTI (ETI) for upper (lower) envelope detection, as explored in [44].

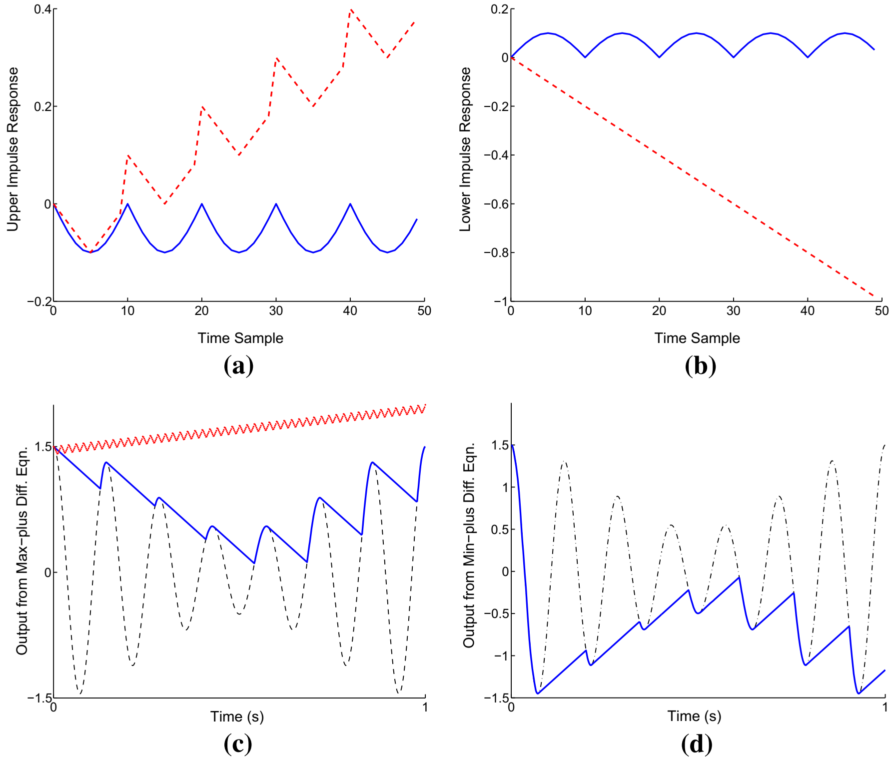

Fig. 2 Responses of DTI or ETI systems described by the recursive max-sum equation (116) or its min-sum version (118); in all cases $m = 0$, $b_0 = 0$. **a** Impulse response (first 50 samples) of a $n = 11$th-order DTI system for two coefficient sequences $\{a_k\}$: in solid line $a_k = -\sin(\pi(k-1)/10)/10$ for $k = 1, \dots, 10$ and $a_{11} = 0$, whereas in dash line $a'_k = (|k-6|-5)/50$ for $k = 1, \dots, 10$ and $a'_{11} = 0.1$. **b** Dual impulse response (first 50 samples) of a 11th-order recursive ETI system for two coefficient sequences: in solid line $a_k = \sin(\pi(k-1)/10)/10$ for $k = 1, \dots, 10$ and $a_{11} = 0$, whereas in dash line $a'_k = (|k-6|-5)/50$ for $k = 1, \dots, 11$. **c** Output signals from two DTI systems whose input (dashed line) are an amplitude-modulated sine. The first output (solid blue line) is from the stable system $y(t) = \max[y(t-1) + a_1, u(t)]$ with $a_1 = -0.008$. The second output (dotted red line) is from the unstable system that generated the unstable impulse response of (**a**). **d** Input signal as in (**c**) and output from the min-sum system $y(t) = \min[y(t-1) - a_1, u(t)]$ (color figure online)

9.1.2 Dynamic programming

The max-sum or min-sum recursive equations can also express various forms of dynamic programming, either of maximizing some gain or minimizing some cost or distance [2,5]. For example, consider (8) and assume that a_{ij} is the transition gain from state i to state j between two consecutive time instants and that $x_i(t)$ represents the maximum possible gain to reach state i in t steps starting from some initial state at $t = 0$. Then (8) with a transposed transition matrix, i.e., the max-sum system

$$\boldsymbol{x}(t) = \boldsymbol{A}^{\mathrm{T}} \boxplus \boldsymbol{x}(t-1), \quad \boldsymbol{x}(0) = [0, -\infty, \dots, -\infty]^{\mathrm{T}}, \tag{119}$$

models a dynamic programming algorithm where, starting from state 1 with zero gain, we move from state to state aiming at solving the above optimization problem by sequentially maximizing the gain. The optimum path can be found by backtracking.

Instead of max-sum, there is also a max-product example of dynamic programming presented in Sect. 9.2.1. Other abstract models of dynamic programming have been studied in [64].

9.1.3 Distance maps and min-plus recursions

The min-sum version of (8) models shortest path problems. Given a 2D rectangular field $f : \mathcal{V} \to \mathbb{R}$ on a grid $\mathcal{V}$ of $M \times N$ pixels, its weighted distance transform is defined by

$$D_f(i, j) = \bigwedge_{(k,\ell)\in\mathcal{V}} d(i-k, j-\ell) + f(k, \ell) \tag{120}$$

where $d(\cdot)$ is the Euclidean distance. For various cases of f, the above distance computation problem is at the heart of several well-known optimization problems [25],[61]. If D_f is available, we can solve the shortest path problem from any point by following the gradient of the distance map.

If f equals q'_S, which is the lower indicator function of a set $S \subseteq \mathcal{V}$ with values 0 on S and $+\infty$ on $\mathcal{V}\backslash S$, then D_f becomes the *distance transform* of the set S:

$$D_S(i, j) = \min_{(k,\ell)\in S} \|i-k, j-\ell\| \tag{121}$$

which measures distances from S out into its containing field. Consider indexing rowwise the 2D rectangular grid $\mathcal{V}$ of $M \times N$ pixels (i, j) as a 1D sequence of points $t = N(i-1) + j$, $i = 1, \ldots, M$, $j = 1, \ldots, N$. A good approximation to the Euclidean distance function $D_S(t)$ is to compute the chamfer distance [11] by propagating a 3×3 mask (8-pixel neighborhood) of local distance steps (a, b).

A serial implementation is an iterative algorithm where the 8-pixel neighborhood is partitioned into two 4-pixel subneighborhoods, and each new array of results sequentially passes through recursive infimal convolutions $y_i(t)$, $i = 1, 2, 3, \ldots$, which for odd i are a forward pass with the submask of Fig. 3a scanning rowwise the 2D field from top to bottom and for even i are a backward pass with the reflected submask in the reverse scanning order. The ith forward pass is described by the min-sum difference equation

$$y_i(t) = [\bigwedge_{k=1}^{N+1} w_k + y_i(t-N+k-2)] \wedge u_{i-1}(t) \tag{122}$$

where $w_1 = b$, $w_2 = a$, $w_3 = b$, $w_{N+1} = a$ and all other w_k are $+\infty$, $u_0 = q'_S$ and $u_i = y_i$ for $i \geq 1$. Its dynamics can be modeled by the min-sum version of the general state equations (6). It admits a state-space model as in (117) with $n = N + 1$ states $x_k(t) = y(t-N+k-2)$, the only differences being that the null elements

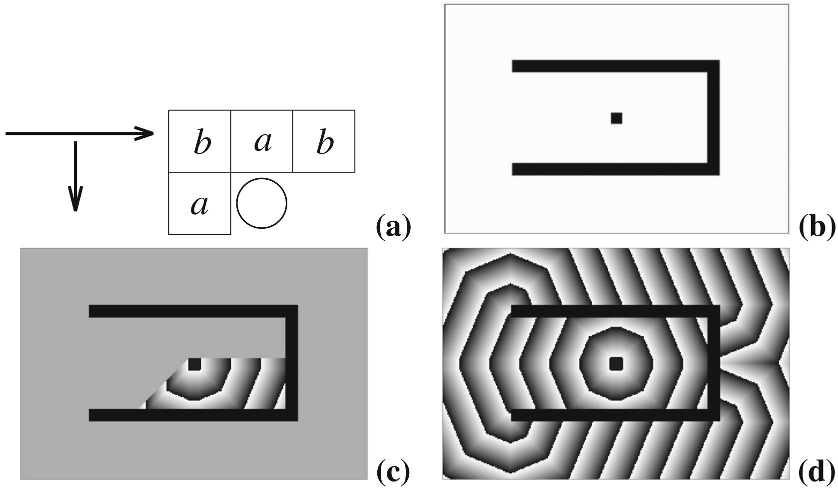

Fig. 3 **a** Coefficient submask for forward pass of sequential distance transform. **b** Source set S and the obstacle wall set W. **c** First (forward) pass of constrained distance transform with steps $(a, b) = (24, 34)/25$. **d** Fourth (backward) pass and final result, shown with gray values modulo a constant

$(-\infty)$ in the sparse system matrices should be replaced with $+\infty$, and all elements in the last row of $\boldsymbol{A}$ and in $\boldsymbol{C}$ are $+\infty$ except at four positions ($k = 1, 2, 3, N + 1$) where they are equal to the corresponding local distances. The source set S could be a small region from which we propagate distances; see Fig. 3b, c. If the field contains impenetrable obstacles (like 'walls') W, distance maps can be produced that account for this impenetrability, and then shortest paths can be found that avoid collision with the walls, which is useful in robotics [63]. This can be done by imposing in each iteration values $+\infty$ at all points of the wall W. The algorithm (122) generally converges to $D_S(t) = \lim y_i(t)$, and the number of required passes is two if there are no obstacles; see Fig. 3d. For a 1D sequence S of points, we need only two passes as the following example illustrates with recursions $y_1(t) = \min[y_1(t-1) + 1, u_0(t)]$ and $y_2(t) = \min[y_2(t+1) + 1, y_1(t)]$:

u_0	∞	0	∞	∞	∞	0	∞	∞	∞	∞	∞	0
y_1	∞	0	1	2	3	0	1	2	3	4	5	0
y_2	1	0	1	2	1	0	1	2	3	2	1	0

Note that both recursive equations are stable min-plus systems.

9.2 Max-product systems

Another class of nonlinear dynamical systems is obtained by using the nonnegative numbers $\mathcal{K} = [0, +\infty]$ as scalars, the standard product $(\times)$ as scalar 'multiplication,' and the following max-product $\boxtimes$ and its dual $\boxtimes'$ as generalized matrix 'multiplications' in (2) and (6):

$$\boldsymbol{C} = \boldsymbol{A} \boxtimes \boldsymbol{B} = [c_{ij}] \ , \ c_{ij} = \bigvee_{k=1}^{n} a_{ik} \times b_{kj} \tag{123}$$

$$\boldsymbol{C} = \boldsymbol{A} \boxtimes' \boldsymbol{B} = [c_{ij}] \ , \ c_{ij} = \bigwedge_{k=1}^{n} a_{ik} \times' b_{kj} \tag{124}$$

The scalar multiplications $\times$ and $\times'$ coincide over $(0, +\infty)$, but $a \times 0 = 0$ and $a \times' (+\infty) = +\infty$ for all $a \in [0, +\infty]$. Henceforth, we shall use the same symbol $\times$ for both scalar operations. Here the scalar arithmetic is based on the max-times clog $([0, \infty], \vee, \wedge, \times)$. This max-product formalism can model dynamical systems whose inputs, states, and outputs are constrained to be nonnegative. Note that there is an isomorphism between the max-sum and the max-product systems because, if we have the following max-product state equations

$$\boldsymbol{x}(t) = \boldsymbol{A} \boxtimes \boldsymbol{x}(t-1) \vee \boldsymbol{B} \boxtimes \boldsymbol{u}(t) \tag{125}$$

and take logarithms of both sides elementwise, we obtain the max-sum equations

$$\log \boldsymbol{x}(t) = \log \boldsymbol{A} \boxplus \log \boldsymbol{x}(t-1) \vee \log \boldsymbol{B} \boxplus \log \boldsymbol{u}(t) \tag{126}$$

Such systems have found applications in speech recognition and other natural language processing tasks using finite-state automata [34,52], in computer vision [25], the max-product algorithm in belief propagation [54] and related probabilistic graphical models used in machine learning [7].

9.2.1 Viterbi algorithm and HMMs

Given a time sequence of observations (feature vectors) $\boldsymbol{O} = (\boldsymbol{o}_t)_{t=0}^{\mathrm{T}}$, a fundamental problem in their statistical modeling using hidden Markov models (HMMs) [55] with n discrete states $\{1, \ldots, n\}$ is to find the best sequence of states $\hat{\boldsymbol{s}} = (s_0, s_1, \ldots, s_T)$ that maximizes the probability $\Pr(\boldsymbol{O}, \boldsymbol{s}|\boldsymbol{\theta})$, where $\boldsymbol{\theta} = ([\pi_i], [a_{ij}], [p_i])$ are the HMM parameters: π_i are the initial state probabilities at $t = 0$, $a_{ij} = \Pr(s_t = j|s_{t-1} = i)$ are state transition probabilities, and $p_i(t)$ are the state-conditional observation probabilities $p(\boldsymbol{o}_t|s_t = i)$ often modeled by Gaussian Mixture models (GMMs). Consider the highest probability of a single partial state sequence ending at state i at time t and accounting for the first $t+1$ observations:

$$x_i(t) = \max_{s_0, \ldots, s_{t-1}} \Pr[s_0, \ldots, s_{t-1}, s_t = i, \boldsymbol{o}_0, \ldots, \boldsymbol{o}_t|\boldsymbol{\theta}] \tag{127}$$

One solution is to use the Viterbi algorithm to find the max global score

$$\hat{P} = \Pr(\boldsymbol{O}, \hat{\boldsymbol{s}}|\boldsymbol{\theta}) = \max_i x_i(T) \tag{128}$$

and then find the optimal state sequence via backtracking. This is essentially dynamic programming and amounts to evolving the following system, for $t = 1, \dots, T$,

$$\begin{aligned} x_i(t) &= \left(\bigvee_{j=1}^n a_{ji} x_j(t-1)\right) \cdot p_i(t) \\ y(t) &= \bigvee_{i=1}^n x_i(t) \end{aligned} \tag{129}$$

with $x_i(0) = \pi_i p_i(0)$. Then, this is a max-product system with matrices $\boldsymbol{A}(t) = [a_{ji}]p_i(t)$, $\boldsymbol{C} = [1, 1, \dots, 1]$ and zero input. The Viterbi score is given by the final output $\hat{P} = y(T)$.

9.2.2 Attention control and multimodal saliencies

Assume a video sequence of audiovisual (AV) events each to be scored with some degree of saliency in [0, 1] where ‘saliency’ is some bottom-up low-level sensory form of attention by a human watching this video. The states x_1, x_2, x_3, x_4 represent time-evolving mono- or multimodal saliencies, where 1 = audio, 2 = visual, 3 = audiovisual, and 4 = nonsalient. Peaks in these saliency trajectories signify important events, which can be automatically detected and produce video summaries that agree well with human attention [24]. The following state equations are a possible max-product dynamical model we have proposed for the evolution of these saliency states [47]:

$$x_i(t) = \left(\bigvee_{j=1}^4 a_{ji} x_j(t-1)\right) \star p_i(t) \vee \left(\bigvee_{j=1}^4 b_{ij} u_j(t)\right) \tag{130}$$

for state $i = 1, 2, 3, 4$. The constants a_{ij} represent state transitions probabilities and $p_i(t) = p(\boldsymbol{o}_t | s_t = i)$ denote the probabilities of observed low-level feature vectors $\boldsymbol{o}_t$ while being at the ith saliency state. We assume that the parameters a_{ij}, b_{ij} and $p_i(t)$ are given.

Given a time sequence of such observations $(\boldsymbol{o}_t)$ one can fit HMMs to these data using maximum likelihood. Then, the first term in the RHS of (130) models the evolution of the Viterbi dynamic programming algorithm (129) used in HMMs. For example, if the inputs $u_i(t)$ are all null, then the single output $y(t) = \bigvee_i x_i(t)$ computes the Viterbi score (128). One main difference of our system (130) with the Viterbi algorithm (129) is that we have the probability-like signals $u_i(t)$ which can act as control inputs coming possibly from previous human attention states or higher-level events, e.g., detected human faces, voice activity, or text semantics. Another difference is that the outputs of the dynamical system can be various min/max combinations of the saliency states of various modalities, e.g., the single output

$$y(t) = c_1 x_1(t) \vee c_2 x_2(t) \vee d_1 u_1(t) \vee d_2 u_2(t) \tag{131}$$

forms a weighted max-product fusion of the audio and visual saliencies as well as the two corresponding inputs. In such modality and input combinations, the max rule can

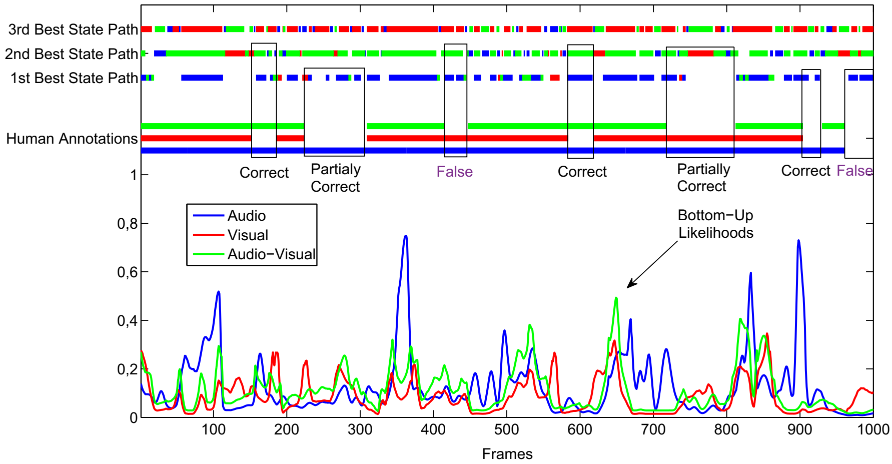

Fig. 4 Evolution of audio (blue), visual (red) and audiovisual (green) bottom-up likelihoods computed from observed features. We also see the human annotations and the 3-Best state paths using the max-product dynamical system (130) with $\star$ being product operation and two control inputs $u_1(t)$ and $u_2(t)$ providing binary information from voice and face detection, respectively (color figure online)

be replaced by min too. A third difference is that the data-controlled probabilities $p_i(t)$ can enter not only via multiplication but also via any commutative binary operation $\star$ that distributes over maximum. If $\star = \max$, then the $p_i(t)$ can be viewed as control inputs. Finally, our CWL theoretical formulation allows us to also compute analytically the responses of such max-product dynamical systems; see Sect. 4.

In our experiments [47], we estimated the state transition probabilities a_{ij} using the EM algorithm on some training data from movie videos. For estimating the observation data probabilities $p_i(t)$, we fitted GMMs to audio and visual feature vectors extracted from the video data at each frame t. Figure 4 shows the results (on testing data from the same movie videos) of various approaches we have initiated to track the joint audiovisual (AV) saliency state and compare it (i) with human annotations, i.e., binary AV saliency manually annotated by a human who observed these movie videos, and (ii) with an AV saliency automatically computed in [24] by fusing saliencies of the audio and visual streams measured from monomodal cues. Our ongoing research goal here is to develop a computational model that can track human attention in the form of audiovisual saliency states based on multimodal sensory inputs. As shown in Fig. 4 and explained numerically in [47], our results using the max-product dynamical system are encouraging; they can track audiovisual saliencies with smaller error than bottom-up feature-based local measurements and can improve with higher-level control input.

9.3 Max-T norm systems and fuzzy Markov chains

There are many types of nonlinear dynamical systems where the elements of the state, input and output vectors represent probabilities or memberships. Examples include probabilistic or fuzzy control systems [1,39,49], fuzzy image convolutions [8,45], as

Table 4 T-norms, conorms and their adjoints

T-norm	Adjoint t-norm	T-conorm	Adjoint t-conorm
$T(a, v)$	$\zeta(a, w)$	$U = T^*(a, w)$	$\eta(a, v)$
$\min(a, v)$	$\max([w \geq a], w)^{\dagger}$	$\max(a, w)$	$\min([v > a], v)$
$a \cdot v$	$\min(w/a, 1)$	$a + w - a \cdot w$	$\max(\frac{v-a}{1-a}, 0)$

$^{\dagger}[P]$ is the Iverson bracket with value 1 (0) if P is true (false)

well as certain types of neural nets with max–min combinations of inputs [39,66]. The dynamics of large classes of such systems can be described by the general model if we restrict the set of scalars to be $\mathcal{K} = [0, 1]$ and use as scalar 'multiplication' $a \star b = T(a, b)$ a *fuzzy intersection norm* [39], i.e., a binary operation $T : [0, 1]^2 \to [0, 1]$ that is i) commutative, ii) associative, iii) increasing and iv) satisfying the boundary condition $T(a, 1) = a$ for all $a \in [0, 1]$. This is also known as 'triangular norm' (t-norm) in statistics. We also require that T is continuous, which makes it a scalar dilation [49]. As dual 'scalar multiplication,' we use a continuous binary operation $U(a, b) = a \star' b$ that satisfies (i)–(iii) and the dual boundary condition $U(a, 0) = a$. This is a *fuzzy union norm* [39], also known as 't-conorm,' and is a scalar erosion on [0, 1].

Choosing in the general lattice dynamical model the above set of scalars and 'multiplications' among them creates the case of max-Tnorm and min-Unorm systems, obtained by replacing in (2) and (6) the general max-$\star$ matrix multiplication and its dual with the following versions:

$$\boldsymbol{C} = \boldsymbol{A} \boxed{\star} \boldsymbol{B} = [c_{ij}] \ , \ c_{ij} = \bigvee_{k=1}^{n} T(a_{ik}, b_{kj}) \tag{132}$$

$$\boldsymbol{C} = \boldsymbol{A} \boxed{\star}' \boldsymbol{B} = [c_{ij}] \ , \ c_{ij} = \bigwedge_{k=1}^{n} U(a_{ik}, b_{kj}) \tag{133}$$

Usually we select $U(a, b) = a \star' b = T^*(a, b)$ where T^* is the conjugate norm obtained via fuzzy complementation:

$$T^*(a, b) = 1 - T(1 - a, 1 - b) \tag{134}$$

then (T, T^*) form a negation duality, but not an adjunction. The most well-studied choice for the T norm and its dual norm T^* are the min and max operations, respectively. Another known case is for T to equal the product operation. Table 4 shows these cases and their adjoints so that (T, ζ) and (η, T^*) are scalar adjunctions. There are also numerous other choices.

An application of the above ideas to state-space description and control of fuzzy dynamical systems is presented in [49]. Further, dynamical systems with states $\boldsymbol{x}(t) \in [0, 1]^n$ and transition rule based on the max–min matrix 'multiplication' ($\star = \min$)

$$\boldsymbol{x}(t+1) = \boldsymbol{A} \boxed{\star} \boldsymbol{x}(t) = \boldsymbol{A}^{(t)} \boxed{\star} \boldsymbol{x}(0), \quad \boldsymbol{A} = \boldsymbol{P}^{\mathrm{T}} \tag{135}$$

where $\boldsymbol{P} = [p_{ij}] \in [0, 1]^{n \times n}$ is the matrix of state transition probabilities or fuzzy relations among states (i, j), have been called *fuzzy Markov chains (FMCs)* in [1] and studied for decision-making. An advantage they have over classical Markov chains (whose transition rule is based on the sum-product matrix multiplication) is that the powers of the transition matrix always reach a stationary solution $\boldsymbol{x}(\infty)$ in a finite number of steps. Namely, the max–min powers of any matrix $\boldsymbol{A}$ either converge in a finite time τ, i.e., $\boldsymbol{A}^{(\tau+1)} = \boldsymbol{A}^{(\tau)}$, or oscillate with a finite period ν after some finite power τ. In the aperiodic case ($\nu = 1$), if the limiting matrix $\boldsymbol{A}^{(\tau)}$ has identical columns, then the stationary solution $\boldsymbol{x}(\infty)$ is independent of the initial state $\boldsymbol{x}(0)$ and the FMC is called *ergodic*.

We can extend these results for more general FMCs by using alternative T-norms, e.g., the product. Specifically, for both cases of Table 4 (i.e., when T is the minimum or product operation on [0, 1]) Theorem 5 applies and in particular (95) always holds. From this, we can deduce the finite convergence properties of generalized FMCs. Further, if $a_{ii} = 1$ for all i, then it follows that $\boldsymbol{A}^{(t)} \leq \boldsymbol{A}^{(t+1)}$ for all $t \geq 1$; hence from (95) we can prove an aperiodic finite convergence since

$$\boldsymbol{\Gamma}(\boldsymbol{A}) = \boldsymbol{A}^{(n)} = \boldsymbol{A}^{(t)} \quad \forall t > n \tag{136}$$

Thus, $\boldsymbol{A} \boxed{\star} \boldsymbol{\Gamma}(\boldsymbol{A}) = \boldsymbol{\Gamma}(\boldsymbol{A})$. This implies that all columns of the metric matrix $\boldsymbol{\Gamma}(\boldsymbol{A})$ are solutions of

$$\boldsymbol{A} \boxed{\star} \boldsymbol{x} = \boldsymbol{x} \tag{137}$$

Such vectors are max-T eigenvectors of $\boldsymbol{A}$ whose principal eigenvalue is $\lambda(\boldsymbol{A}) = 1$ and provide stationary solutions of the FMC. As a numerical example, consider the transition matrix $\boldsymbol{A}$ and its powers of a max–min FMC:

$$\boldsymbol{A} = \begin{bmatrix} 1 & 0.4 & 0 \\ 0.3 & 1 & 0.5 \\ 0.7 & 0.2 & 1 \end{bmatrix} \leq \boldsymbol{A}^{(2)} = \boldsymbol{A}^{(3)} = \boldsymbol{\Gamma}(\boldsymbol{A}) = \begin{bmatrix} 1 & 0.4 & 0.4 \\ 0.5 & 1 & 0.5 \\ 0.7 & 0.4 & 1 \end{bmatrix} \tag{138}$$

The columns of $\boldsymbol{\Gamma}(\boldsymbol{A})$ provide stationary solutions of this FMC, e.g.,

$$\boldsymbol{A} \boxed{\star} [1, 0.5, 0.7]^{\mathrm{T}} = [1, 0.5, 0.7]^{\mathrm{T}}. \tag{139}$$

10 Conclusions

In this work, we have a developed a unified theory of nonlinear dynamical systems of the max-$\star$ type and their dual min-$\star'$ type over nonlinear vector and signal spaces which we call complete weighted lattices (CWLs). Special cases include max-sum or min-sum systems encountered in discrete event systems and shortest path problems, max-product systems in statistical inference like the Viterbi algorithm, and the max-fuzzy-norms systems encountered in certain types of neural nets and fuzzy control. We have studied several control-theoretical and signal processing aspects of such systems, both by using CWLs for shorter proofs of known cases and by extending the theory to more general cases. Further, we have also outlined several application areas that are either new or not often encountered in the literature, which has emphasized so far the max-plus case and its application to discrete events systems; examples include state-space representation and stability analysis of geometric filtering, distance maps, fuzzy Markov chains, a generalized Viterbi algorithm for HMMs with control inputs and its application to tracking salient events in multimodal videos.

Overall, the unified formulation of the above systems and the corresponding CWL framework provide several advantages over minimax algebra which include: capability of handling both finite- and infinite-dimensional cases; coexistence over the same space of the max-$\star$ and the dual min-$\star'$ systems; lattice monotone operators that can represent matrix–vector multiplications in both state-space and sup/inf input–output signal convolutions; lattice adjunctions (pairs of dual operators) that yield optimal solutions to max-$\star$ and min-$\star'$ equations via lattice projections.

Acknowledgements The author wishes to thank the anonymous Reviewers for their constructive comments. He also wishes to thank Petros Koutras at NTUA CVSP laboratory for producing Fig. 4 and Anastasios Tsiamis at the University of Pennsylvania for Example 6.

References

1. Avrachenkov KE, Sanchez E (2002) Fuzzy Markov chains and decision-making. Fuzzy Optim Decis Mak 1:143–159
2. Baccelli F, Cohen G, Olsder GJ, Quadrat J-P (1992) Synchronization and linearity: an algebra for discrete event systems (web edition, 2001). Wiley, New York
3. Bellman R, Karush W (1961) On a new functional transform in analysis: the maximum transform. Bull Am Math Soc 67:501–503
4. Bellman R, Karush W (1963) On the maximum transform. J Math Anal Appl 6:67–74
5. Bertsekas DP (2012) Dynamic programming and optimal control. Athena Scientific, Belmont (vol. I, 3rd ed., vol. II, 4th ed.)
6. Birkhoff G (1967) Lattice theory. American Mathematical Society, Providence, RI
7. Bishop CM (2006) Pattern recognition and machine learning. Springer, Berlin
8. Bloch I, Maitre H (1995) Fuzzy mathematical morphologies: a comparative study. Pattern Recognit 9(28):1341–1387
9. Blyth TS (2005) Lattices and ordered algebraic structures. Springer, Berlin
10. Blyth TS, Janowitz MF (1972) Residuation theory. Pergamon Press, Oxford
11. Borgefors G (1984) Distance transformations in arbitrary dimensions. Comput Vis Graph Image Process 27:321–345
12. Brockett RW (1970) Finite-dimensional linear systems. Wiley, New York
13. Brockett RW (1994) Language driven hybrid systems. In: Proceedings of 33rd conference on decision and control
14. Brogan WL (1974) Modern control theory. Quantum Publishers, New York
15. Butkovič P (2010) Max-linear systems: theory and algorithms. Springer, Berlin
16. Cassandras CG, Lafortune S (1999) Introduction to discrete event systems. Kluwer, Dordrecht
17. Charisopoulos V, Maragos P (2017) Morphological perceptrons: geometry and training algorithms. In: Proceedings international symposium on mathematical morphology (ISMM-2017), Fontainebleau, France
18. Cohen G, Dubois D, Quadrat JP, Viot M (1985) A linear system theoretic view of discrete event processes and its use for performance evaluation in manufacturing. IEEE Trans Autom Control 30:210–220

19. Cohen G, Gaubert S, Quadrat JP (2004) Duality and separation theorems in idempotent semimodules. Linear Alegbra Appl 379:395–422
20. Cohen G, Moller P, Quadrat JP, Viot M (1989) Algebraic tools for the performance evaluation of discrete event systems. Proc IEEE 77:39–58
21. Cuninghame-Green R (1979) Minimax algebra. Springer, Berlin
22. Cuninghame-Green RA, Butkovic P (2003) The equation Ax=By over (max, +). Theor Comput Sci 293:3–12
23. Doustmohammadi A, Kamen EW (1995) Direct generation of event-timing equations for generalized flow shop systems. In: Modeling, simulation, and control technologies for manufacturing, volume 2596 of proceedings of SPIE, pp 50–62
24. Evangelopoulos G, Zlatintsi A, Potamianos A, Maragos P, Rapantzikos K, Skoumas G, Avrithis Y (2013) Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention. IEEE Trans Multim 15(7):1553–1568
25. Felzenszwalb PF, Huttenlocher DP (2004) Distance transforms of sampled functions. Technical Report TR2004-1963, Cornell University
26. Gaubert S, Katz RD (2007) The Minkowski theorem for max-plus convex sets. Linear Alegbra Appl 421:356–369
27. Gazarik MJ, Kamen EW (1999) Reachability and observability of linear systems over max-plus. Kybernetika 35(1):2–12
28. Gondran M, Minoux M (2008) Graphs, dioids and semirings: new models and algorithms. Springer, Berlin
29. Hardouin L, Cottenceau B, Lhommeau M, Le Corronc E (2009) Interval systems over idempotent semiring. Linear Alegbra Appl 431(5–7):855–862
30. Heidergott B, Olsder GJ, van der Woude J (2006) Max plus at work: modeling and analysis of synchronized systems: a course on max-plus algebra and its applications. Princeton University Press, Princeton
31. Heijmans HJAM (1994) Morphological image operators. Academic Press, Boston
32. Heijmans HJAM, Ronse C (1990) The algebraic basis of mathematical morphology. Part I: dilations and erosions. Comput Vis Graph Image Process 50:245–295
33. Ho Y-C (ed) (1992) Discrete event dynamic systems: analyzing complexity and performance in the modern world. IEEE Press, New York
34. Hori T, Nakamura A (2013) Speech recognition algorithms using weighted finite-state transducers. Morgan & Claypool, San Rafael
35. Kaburlasos VG, Petridis V (2000) Fuzzy lattice neurocomputing (FLN) models. Neural Netw 13:1145–1169
36. Kailath T (1980) Linear systems. Prentice-Hall, Englewood Cliffs
37. Kamen EW (1993) An equation-based approach to the control of discrete even systems with applications to manufacturing. In: Proceedings of international conference on control theory and its applications, Jerusalem, Israel
38. Kamen EW, Doustmohammadi A (1994) Modeling and stability of production lines based on arrival-to-departure delays. In: Proceedings 33rd conference on decision and control
39. Klir GJ, Yuan B (1995) Fuzzy sets and fuzzy logic: theory and applications. Prentice-Hall, Englewood Cliffs
40. Lahaye S, Boimond J-L, Hardouin L (2004) Linear periodic systems over dioids. Discrete Event Dyn Syst 14:133–152
41. Litvinov GL, Maslov VP, Shpiz GB (2001) Idempotent functional analysis: an algebraic approach. Math Notes 69(5):696–729
42. Litvinov GL, Sobolevskii AN (2001) Idempotent interval analysis and optimization problems. Reliab Comput 7(5):353–377
43. Lucet Y (2010) What shape is your conjugate? A survey of computational convex analysis and its applications. SIAM Rev 52(3):505–542
44. Maragos P (1994) Morphological systems: slope transforms and max–min difference and differential equations. Signal Process 38:57–77
45. Maragos P (2005) Lattice image processing: a unification of morphological and fuzzy algebraic systems. J Math Imaging Vis 22:333–353

46. Maragos P (2013) Representations for morphological image operators and analogies with linear operators. In: Hawkes PW (ed) Advances in imaging and electron physics, vol 177. Academic Press, London, pp 45–187
47. Maragos P, Koutras P (2015) Max-product dynamical systems and applications to audio-visual salient event detection in videos. In: Proceedings IEEE international conference on acoustics speech and signal processing (ICASSP)
48. Maragos P, Schafer RW (1990) Morphological systems for multidimensional signal processing. Proc IEEE 78:690–710
49. Maragos P, Stamou G, Tzafestas SG (2000) A lattice control model of fuzzy dynamical systems in state-space. In: Goutsias J, Vincent L, Bloomberg D (eds) Mathematical morphology and its application to image and signal processing. Kluwer, Dordrecht
50. Maslov VP (1987) On a new superposition principle for optimization problems. Uspekhi Mat Nauk [Russ Math Surv] 42(3):39–48
51. McEneaney WM (2006) Max-plus methods for nonlinear control and estimation. Birkhauser, Boston
52. Mohri M, Pereira F, Ripley M (2002) Weighted finite-state transducers in speech recognition. Comput Speech Lang 16:69–88
53. Oppenheim AV, Schafer RW (1989) Discrete-time signal processing. Prentice-Hall, Englewood Cliffs, NJ
54. Pearl J (1988) Probabilistic reasoning in intelligent systems: networks of plausible inference. Morgan Kaufman Publishers, Los Altos
55. Rabiner L, Juang B-H (1993) Fundamentals of speech recognition. Prentice-Hall, Englewood Cliffs
56. Ritter GX, Sussner P, Diaz de Leon JL (1998) Morphological associative memories. IEEE Trans Neural Netw 9(2):281–293
57. Ritter GX, Urcid G (2003) Lattice algebra approach to single-neuron computation. IEEE Trans Neural Netw 14(2):282–295
58. Rockafellar RT (1970) Convex analysis. Princeton University Press, Princeton
59. Serra J (ed) (1988) Image analysis and mathematical morphology, volume 2: theoretical advances. Academic Press, London
60. Sternberg SR (1986) Grayscale morphology. Comput Vis Graph Image Process 35:333–355
61. Tsitsiklis JN (1995) Efficient algorithms for globally optimal trajectories. IEEE Trans Autom Control 40(9):1528–1538
62. van den Boom TJJ, De Schutter B (2012) Modeling and control of switching max-plus-linear systems with random and deterministic switching. Discrete Event Dyn Syst 22:293–332
63. Verbeek P, Dorst L, Verwer B, Groen F (1986) Collision avoidance and path finding through constrained distance transformation in robot state space. In: Proceedings of international conference on intelligent autonomous systems, Amsterdam
64. Verdu S, Poor HV (1987) Abstract dynamic programming models under commutativity conditions. SIAM J Control Optim 25(4):990–1006
65. Wagneur E (1991) Moduloids and pseudomodules–1. Dimension theory. Discrete Math 98:57–73
66. Yang P-F, Maragos P (1995) Min–max classifiers: learnability, design and application. Pattern Recognit 28(6):879–899