Applications of the duality of min-plus and max-plus network calculus

Jörg Liebeherr

University of Toronto

(WoNeCa 2018)
Background for this talk


Available from my home page (see: Publications)
Originally ....

... I wanted to write an elementary introduction to max-plus network calculus for a course,

- relate it to the min-plus version, and
- discuss applications to scheduling with rate guarantees

• This was supposed to be easy since:
  - The \((\min, +)\)- and \((\max, +)\)-dioids are isomorphic
  - Operations of the min-plus and max-plus network calculus are well-understood
  - Many have worked with concepts in both algebras
**Min-Plus and Max-Plus Network Calculus**

- **Min-plus**: Arrival, departures, service are functions of time.
- **Max-plus**: Arrival, departures, service are functions of space.
- Functions are related by a reflection at the diagonal!
Remark 6.2.7. We note that not every result in the min-plus algebra can be extended here. For example, a concatenation of the minimal $g_1$-regulator and the minimal $g_2$-regulator is not the minimal $g_1 \circ g_2$-regulator in general.

More specifically, there is not an exact correspondence between the set of flows that are $g$-regular on one hand, and that are $\sigma$-smooth on the other. We explain why with an example.

In the sequel we will restrict our exposition to min-plus systems theory and only use the max-plus approach where it is particularly useful. We note, however, that many concepts can be mirrored in the max-plus algebra.
## Notation

<table>
<thead>
<tr>
<th>Min-plus</th>
<th>Max-plus</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>A(t)</strong></td>
<td><strong>T_A(ν)</strong></td>
</tr>
<tr>
<td>Arrivals until time t</td>
<td>Arrival time of bit  ν</td>
</tr>
<tr>
<td><strong>D(t)</strong></td>
<td><strong>T_D(ν)</strong></td>
</tr>
<tr>
<td>Departures until time t</td>
<td>Departure time of bit  ν</td>
</tr>
<tr>
<td><strong>W(t)</strong></td>
<td><strong>W(ν)</strong></td>
</tr>
<tr>
<td>Virtual delay at time t</td>
<td>Delay of bit  ν:</td>
</tr>
<tr>
<td><strong>B(t)</strong></td>
<td><strong>B^a(ν)</strong></td>
</tr>
<tr>
<td>Backlog at time t</td>
<td>Backlog at arrival of  ν</td>
</tr>
<tr>
<td><strong>B^d(ν)</strong></td>
<td>Backlog at departure of  ν</td>
</tr>
<tr>
<td><strong>S(t)</strong></td>
<td><strong>γ_S(ν)</strong></td>
</tr>
<tr>
<td>Service curve</td>
<td>Service curve</td>
</tr>
<tr>
<td><strong>E(t)</strong></td>
<td><strong>λ_E(ν)</strong></td>
</tr>
<tr>
<td>Envelope (‘arrival curve’)</td>
<td>Envelope (‘arrival curve’)</td>
</tr>
<tr>
<td><strong>∅, ⊗</strong></td>
<td>(De-) Convolution</td>
</tr>
<tr>
<td><strong>F ∈ F_o</strong></td>
<td>(De-) Convolution</td>
</tr>
<tr>
<td>ƒ is left-continuous, non-decreasing,</td>
<td></td>
</tr>
<tr>
<td>( F(t) = 0 ) if ( t \leq 0 )</td>
<td>( f ) is right-continuous, non-decreasing,</td>
</tr>
<tr>
<td></td>
<td>( F(t) = -\infty ) if ( t &lt; 0 )</td>
</tr>
</tbody>
</table>
Buffered Link

- Work-conserving link with fixed rate $C$

\[ A(t) \quad \text{Arrivals} \quad C \quad \text{Departures} \quad D(t) \]

- Offers an exact (min-plus) service curve: $S(t) = Ct$
  such that $D(t) = A \odot S(t)$

- The corresponding max-plus service curve should be: $\gamma S(\nu) = \frac{\nu}{C}$
  with $T_D(\nu) = T_A \odot \gamma S(\nu)$
Buffered Link

\[ T^p_A(n) \text{ Arrival time of } n\text{-th packet} \]

\[ T^p_D(n) \text{ Departure time of } n\text{-th packet} \]

\[ \ell_n \text{ Size of } n\text{-th packet (in bits)} \]

\[ L_n = \sum_{j=1}^{n} \ell_n \]
Recursion for Departure time of $n$th Packet

\[
T^p_D(n) = \max\left\{T^p_A(n), T^p_D(n - 1)\right\} + \frac{\ell_n}{C}
\]

\[
= \max\left\{T^p_A(n) + \frac{\ell_n}{C}, T^p_A(n - 1) + \frac{\ell_{n-1} + \ell_n}{C} \right\}, \ldots
\]

\[
\ldots, T^p_A(1) + \frac{\ell_1 + \ldots + \ell_n}{C}\}
\]

\[
= \max_{0 \leq k \leq n-1}\left\{T^p_A(n - k) + \frac{\ell_{n-k} + \ldots + \ell_n}{C}\right\}
\]

with $T^p_D(0) = 0$. 
We number the bits of the packets

\[0, 1, \ldots, \ell_1 - 1, \quad \ell_1, \ldots, L_2 - 1, \quad \ldots \quad , \quad L_{n-1}, \ldots, L_n - 1\]

Packet 1  |  Packet 2  |  Packet n

Departure of bit $\nu$:

\[T_D(\nu) = \max \left\{ T_A(\nu), T_D(\nu - 1) \right\} + \frac{1}{C}\]

\[= \max \left\{ T_A(\nu) + \frac{1}{C}, T_A(\nu - 1) + \frac{2}{C}, \ldots, T_A(0) + \frac{\nu}{C} \right\}\]

\[= \max_{\kappa=0,1,\ldots,n} \left\{ T_A(\nu - \kappa) + \frac{\kappa + 1}{C} \right\}\]
Bit-level View of Packets

With

\[
F \bigotimes G(\nu) = \max_{\kappa=0,1,\ldots,\nu} \{ F(\nu - \kappa) + G(\kappa) \},
\]

we get either

\[
T_D(\nu) = T_A \bigotimes \gamma_S(\nu) \quad \text{with} \quad \gamma_S(\nu) = \frac{\nu+1}{C}
\]

or

\[
T_D(\nu) = T_A \bigotimes \gamma'_S(\nu) + \frac{1}{C} \quad \text{with} \quad \gamma'_S(\nu) = \frac{\nu}{C}
\]
Towards a Continuous-Space View

If we measure traffic in $\frac{1}{k}$-th of a bit:

$$T_D(\nu) = \max\{T_A(\nu), T_D(\nu - \frac{1}{k})\} + \frac{1}{kC}$$

$$= \max_{\kappa=0, \frac{1}{k}, \frac{2}{k}, \ldots, \nu} \left\{ T_A(\nu - \kappa) + \frac{\kappa + \frac{1}{k}}{C} \right\}$$

$$= T_A \otimes \gamma_S(\nu) + \frac{1}{k} \quad \text{with} \quad \gamma_S(\nu) = \frac{\nu}{C}$$

For $k \to \infty$:

$$T_D(\nu) = T_A \otimes \gamma_S(\nu) \quad \text{with} \quad \gamma_S(\nu) = \frac{\nu}{C}$$
Continuous-space View of Packets

Viewing bits as real numbers:

- **Packet 1**: $0 \leq \nu < \ell_1$
- **Packet 2**: $\ell_1 \leq \nu < L_2$
- **Packet $n$**: $L_{n-1} \leq \nu < L_n$


\[
T_D(L_n^-) = \sup_{0 \leq \kappa \leq L_n^-} \left\{ T_A(L_n^- - \kappa) + \frac{\kappa}{C} \right\}
= T_A \otimes \gamma(L_n^-)
\]

with $\gamma_S(\nu) = \frac{\nu}{C}$
Continuous-space view results in: $S(t) = Ct \leftrightarrow \gamma_S(\nu) = \frac{\nu}{C}$

In a packet-level or bit-level view:

- ‘Extra term’ $\frac{\ell_n}{C}$ for packets (or $\frac{1}{C}$ for bits) reflects a packetization (or ‘bit’-ization)
- ‘Extra term’ is the root cause for reported discrepancies between min-plus and max-plus network calculus

**Next:** Continuous-space max-plus NC and continuous-time min-plus NC are isomorphic $\Rightarrow$ Pseudo-inverse functions
Motivation for Pseudo-inverses

- $A$ and the $T_A$ are diagonal reflections of each other
- If functions are continuous and strictly increasing, diagonal reflection are the inverses
- Since $A$ and $T_A$ are neither, inverse functions do not exist
  $\implies$ Pseudo-inverse functions
Pseudo-inverses

For a non-decreasing function $F$:

- **Lower pseudo-inverse:**
  \[ F^\downarrow(y) = \inf \{ x \mid F(x) \geq y \} = \sup \{ x \mid F(x) < y \} \]

- **Upper pseudo-inverse:**
  \[ F^\uparrow(y) = \sup \{ x \mid F(x) \leq y \} = \inf \{ x \mid F(x) > y \} \]
Pseudo-inverses

**Lower pseudo-inverse**

**Upper pseudo-inverse**
Properties of pseudo-inverses

For non-decreasing functions $F$ and $G$:

1. $F \downarrow$ and $F \uparrow$ are non-decreasing
2. $F \downarrow$ is left-continuous and $F \uparrow$ is right-continuous
3. $F$ is left-continuous $\Rightarrow F = (F \uparrow) \downarrow$
4. $F$ is right-continuous $\Rightarrow F = (F \downarrow) \uparrow$
5. Order-reversing: $F \geq G \Rightarrow F \uparrow \leq G \uparrow$, $F \downarrow \leq G \downarrow$

\[
A = (A \uparrow) \downarrow = T_A \downarrow \\
T_A = (T_A \downarrow) \uparrow = A \uparrow
\]
Mapping functions between time and space domain

1. \( F \in \mathcal{F}_o \Rightarrow F^\uparrow \in \mathcal{T}_o \)
2. \( F \in \mathcal{T}_o \Rightarrow F^\downarrow \in \mathcal{F}_o \)
3. \( \delta^\uparrow = \bar{\delta} \) and \( \delta^\downarrow = \delta \).
Mapping between min-plus and max-plus algebras

Min-plus $\rightarrow$ Max-plus:

1. $(F \land G)^\uparrow = F^\uparrow \lor G^\uparrow$
2. $(F \otimes G)^\uparrow = F^\uparrow \otimes G^\uparrow$
3. $(F \oslash G)^\uparrow = F^\uparrow \oslash G^\uparrow$
4. $(F + G)^\uparrow(\nu) = \inf_{0 \leq \kappa \leq \nu} \max\{F^\uparrow(\kappa), G^\uparrow(\nu - \kappa)\}$
5. $F \in \mathcal{F}_o$ subadditive $\Rightarrow F^\uparrow$ superadditive

Max-plus $\rightarrow$ Min-plus:

1. $(F \lor G)^\downarrow = F^\downarrow \land G^\downarrow$.
2. $\ldots$
Mapping traffic envelopes

Notation:
\[
\begin{cases}
A \sim E \quad & E \\
T_A \sim \lambda_E \quad & \lambda_E
\end{cases}
\] is a \{\text{min-plus}\} \text{ traffic envelope for } \begin{pmatrix} A \\ T_A \end{pmatrix}

\hline
1. \quad A \sim E \implies A^\uparrow \sim E^\uparrow \\
2. \quad T_A \sim \lambda_E \implies T_A^\downarrow \sim \lambda_E^\downarrow
\hline

Example: Token Bucket

\[
E(t) = b + rt \quad \Rightarrow \quad E^\uparrow(\nu) = \left[\frac{\nu - b}{r}\right]^+
\]
Mapping service curves

1. \( D = A \otimes S \Rightarrow D^\dagger = A^\dagger \otimes S^\dagger \)
2. \( D \geq A \otimes S \Rightarrow D^\dagger \leq A^\dagger \otimes S^\dagger \)
3. \( D \leq A \otimes S \Rightarrow D^\dagger \geq A^\dagger \otimes S^\dagger \)

1. \( T_D = T_A \otimes \gamma_S \Rightarrow T_D^\dagger = T_A^\dagger \otimes \gamma_S^\dagger \)
2. \( T_D \leq T_A \otimes \gamma_S \Rightarrow T_D^\dagger \geq T_A^\dagger \otimes \gamma_S^\dagger \)
3. \( T_D \geq T_A \otimes \gamma_S \Rightarrow T_D^\dagger \leq T_A^\dagger \otimes \gamma_S^\dagger \)

Example: Latency-rate service curve

\[
S(t) = R(t - T)I_{t > T} \Rightarrow \quad S^\dagger(\nu) = \begin{cases} 
-\infty, & \text{if } \nu < 0, \\
\frac{\nu}{R} + T, & \text{if } \nu \geq 0.
\end{cases}
\]

Example: Residual service curve

\[
S(t) = [Ct - E_c(t)]^+ \Rightarrow \quad S^\dagger(\nu) = \frac{1}{C} \left( \inf \left\{ x \mid \lambda_c(x) \geq \frac{x + \nu}{C} \right\} + \nu \right)
\]
Why do we care?

- Backlog easier with min-plus: \( B(t) = A(t) - D(t) \)
- Delay is easier with max-plus: \( W(\nu) = T_D(\nu) - T_A(\nu) \)

- **Aggregate of flows:**
  \[
  A(t) = \sum_{j=1}^{N} A_j(t) \quad (\text{min-plus})
  \]
  \[
  T_A(\nu) = \inf_{\nu_1, \ldots, \nu_N} \max_{j=1, \ldots, N} T_{A_j}(\nu_j) \quad (\text{max-plus})
  \]

Min-plus and max-plus network calculus are complementary:

- Capacity provisioning is easier with min-plus network calculus
- Traffic algorithms are easier in a max-plus view
Example: SCED

- **Computing timestamps:** Deadline computation at a SCED scheduler:
  \[ D\ell(\nu) = T_A \square \gamma_S(\nu) \]  
  \[ D\ell(t) = \sup\{x | A \square S(x) \leq A(t)\} \]  
  \[ = (A \square S)^\dagger(A(t)) \]

- **Schedulability:** Condition for SCED schedulability:
  \[ \inf_{\nu = \nu_1 + \ldots + \nu_N} \max_{j=1,\ldots,N} \lambda_{E_j} \square \gamma_{S_j}(\nu_j) \geq \frac{\nu}{C}, \quad \forall \nu \geq 0 \]
  \[ \sum_{j=1}^{N} E_j \square S_j(t) \leq Ct, \quad \forall t \geq 0 \]  
  (max-plus) vs. (min-plus)
Everything maps nicely, right?
Not quite!
Backlog and Delay

- Backlog and delay cannot be mapped exactly with pseudo-inverses.
- We can only provide bounds, e.g.,
  \[ B^a(A(t)) \leq B(t) \leq B^d(D(t)) \]

  which can be quite loose:

- Good news: If \( A \) and \( D \) are continuous at \( T_A(\nu) \) then
  \[ B(T_A(\nu)) = B^a(\nu) \]
Busy periods and busy sequences

• In general, busy periods cannot be described using expressions of the max-plus network calculus

• We define the concept of **busy sequence** as a maximal sequence of bits with non-zero delays

• Busy sequence also helps with defining a strict max-plus service curve
Problems with busy periods and sequences

- A single busy periods may cover multiple busy sequences
- A single busy sequence may cover multiple busy periods
Summary

- Clarification of the relationship between min-plus and max-plus network calculus
  - Dispenses with the frequently made assumption of constant packet sizes for a max-plus analysis

- Now: Can switch between a min-plus or max-plus viewpoint in the same analysis

- Filled a few holes in the max-plus literature, e.g.,
  - busy sequence
  - strict max-plus service curve
  - adaptive max-plus service curve
Supplemental Slides
A strict max-plus service curve $\gamma_S \in \mathcal{T}_o$ satisfies for all $\nu$ and $\mu$ in the same busy sequence, it holds that

$$TD(\nu) - TD(\mu) \leq \gamma_S(\nu - \mu), \text{ if } \nu < \mu,$$

and $$TD(\nu) - TA(\mu) \leq \gamma_S(\nu - \mu), \text{ if } \nu = \mu.$$
An adaptive max-plus service curve $\gamma_S$ for a network element satisfies for all $\nu \geq 0$,

$$T_D(\nu) \leq \inf_{\mu \leq \nu} \left\{ \max \left[ T_D(\mu) + \gamma_S(\nu - \mu), T_A \bigotimes_\mu \gamma_S(\nu) \right] \right\}.$$ 

where

$$F \bigotimes_\mu G(\nu) = \sup_{\mu \leq \kappa \leq \nu} \left\{ F(\kappa) + G(\nu - \kappa) \right\}$$
- Chapter 6 in Chang’s Book
- Let $F$ and $G$ be non-decreasing real-valued functions, and $H(n)$ a non-decreasing integer-valued function:

\[
F \odot_H G(n) = \max_{0 \leq k \leq n} \{F(k) + G(H(n) - H(k))\},
\]

- Setting $\ell'_n = \ell_{n+1}$ and $L'(n) = \sum_{k=0}^{n-1} \ell'_k$ the output of the buffered link is

\[
T^p_D(n + 1) = \tau \odot_{L'} \gamma_S(n) + \frac{\ell'_n}{C}
\]

with $\gamma_S(\nu) = \frac{\nu}{C}$.
- If $\ell_n = 1$, operations $\lor$ and $\odot_{L'}$ yield a dioid.
Mappings of Dioids

- See Chapter 4, in “Synchronization and Linearity: ...”.
- Applies residuation theory for lattices to establish isomorphisms between dioids
- Terminology:

  - residual  $\rightarrow$ upper pseudo-inverse
  - dual residual  $\rightarrow$ lower pseudo-inverse
  - isotone mapping  $\rightarrow$ non-decreasing function
  - isotone and upper semi-continuous  $\rightarrow$ right-continuous
  - isotone and lower semi-continuous  $\rightarrow$ left-continuous
  - residuated mapping  $\rightarrow$ left-continuous and non-decreasing function
  - dual residuated mapping  $\rightarrow$ right-continuous and non-decreasing function
Lindley equation

\[ \bar{W}_n = \max \{0, \bar{W}_{n-1} + S_{n-1} - A_{n-1}\}, \]

with

- \( \bar{W}_n \) queueing time of \( n \)-th packet,
- \( S_{n-1} \) service time of \((n - 1)\)-th packet, and
- \( A_{n-1} \) is time between arrivals of packets \((n - 1)\) and \( n \)

- With \( W_n = \bar{W}_n + S_n \) we can rewrite Lindley equation as
  \[ W_n = \max \{0, W_{n-1} - A_{n-1}\} + S_n. \]

- Since \( W_n = T_{DP}^p(n) - T_{AP}^p(n) \) we can write
  \[ T_{DP}^p(n) = W_n + T_{AP}^p(n) \]
  \[ = \max \{T_{AP}^p(n), T_{DP}^p(n - 1)\} + \frac{\ell_n}{C} \]