Multivariate subdivision and a simple formula

Tomas Sauer
Lehrstuhl für Numerische Mathematik
Justus–Liebig–Universität Gießen
Tomas.Sauer@math.uni-giessen.de

20.3.2008
Multivariate subdivision and a simple formula

The curse of dimension

Tomas Sauer
Lehrstuhl für Numerische Mathematik
Justus–Liebig–Universität Gießen
Tomas.Sauer@math.uni-giessen.de

20.3.2008
Multivariate subdivision and a simple formula

The curse of dimension and the cursed dimension

Tomas Sauer
Lehrstuhl für Numerische Mathematik
Justus–Liebig–Universität Gießen
Tomas.Sauer@math.uni-giessen.de

20.3.2008
Dimensions and Curses

- “Curse of dimension”:

   What is so bad?
Dimensions and Curses

“Curse of dimension”: Computational complexity increases exponentially with the number of variables.
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases exponentially with the number of variables.

- Example: Grid with stepwidth $h$ in $[0, 1]^S$
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases \textit{exponentially} with the number of variables.

- \textbf{Example}: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases exponentially with the number of variables.

- Example: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.

- But: there are conceptional difficulties:
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases exponentially with the number of variables.

- Example: Grid with stepwidth $h$ in $[0,1]^s$ has $h^{-s}$ grid points.

- But: there are conceptional difficulties:
  - Polynomials usually cannot be factorized.
Dimensions and Curses

- 
  - "Curse of dimension": Computational complexity increases \textit{exponentially} with the number of variables.

- \textbf{Example}: Grid with stepwidth $h$ in $[0,1]^s$ has $h^{-s}$ grid points.

- \textbf{But}: there are \textit{conceptional} difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
“Curse of dimension”: Computational complexity increases exponentially with the number of variables.

Example: Grid with stepwidth $h$ in $[0,1]^s$ has $h^{-s}$ grid points.

But: there are conceptional difficulties:

- Polynomials usually cannot be factorized.
- Polynomials form no principal ideal ring.
- Relevance of geometry.

What is so bad?
Dimensions and Curses

“Curse of dimension”: Computational complexity increases exponentially with the number of variables.

Example: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.

But: there are conceptional difficulties:

- Polynomials usually cannot be factorized.
- Polynomials form no principal ideal ring.
- Relevance of geometry.

Problems:
Dimensions and Curses

- "Curse of dimension": Computational complexity increases exponentially with the number of variables.

- Example: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.

- But: there are conceptional difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
  - Relevance of geometry.

Problems: approximation order, rate of convergence, smoothness etc.
Dimensions and Curses

- "Curse of dimension": Computational complexity increases exponentially with the number of variables.

- Example: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.

- But: there are conceptional difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
  - Relevance of geometry.

Problems: approximation order, rate of convergence, smoothness etc. are related to polynomials!
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases exponentially with the number of variables.

- **Example:** Grid with stepwidth $h$ in $[0,1]^s$ has $h^{-s}$ grid points.

- **But:** there are conceptional difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
  - Relevance of geometry.

**Problems:** approximation order, rate of convergence, smoothness etc. are related to polynomials!

“Locally, polynomials are kings” (C. de Boor)
Dimensions and Curses

- “Curse of dimension”: Computational complexity increases exponentially with the number of variables.

- **Example**: Grid with stepwidth $h$ in $[0,1]^s$ has $h^{-s}$ grid points.

- **But**: there are conceptional difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
  - Relevance of geometry.

**Problems**: approximation order, rate of convergence, smoothness etc. are related to polynomials!

“Locally, polynomials are kings” (C. de Boor)

- Real curse of dimension
Dimensions and Curses

- "Curse of dimension": Computational complexity increases *exponentially* with the number of variables.

- **Example**: Grid with stepwidth $h$ in $[0, 1]^s$ has $h^{-s}$ grid points.

- **But**: there are *conceptional* difficulties:
  - Polynomials usually cannot be factorized.
  - Polynomials form no principal ideal ring.
  - Relevance of geometry.

**Problems**: approximation order, rate of convergence, smoothness etc. are related to polynomials!

  “Locally, polynomials are kings” (C. de Boor)

- Real curse of dimension – cursed dimension!
Refinable functions

**Definition.** \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( \alpha : \mathbb{Z} \to \mathbb{R} \).
Refinable functions

**Definition.** \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi(2 \cdot -k). \tag{\star}
\]
Refinable functions

Definition. \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k).
\]  

(\text{\textquoteleft\textquoteleft }Refinement \text{\textquoteright\textquoteright\ equation\textquoteright\textquoteright\textquoteright},)
Refinable functions

Definition. \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k) .
\]

\( (*) \)

"Refinement equation", classical . . .
Refinable functions

Definition. \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k) .
\]

\( \star \)

"Refinement equation", classical . . .

\( \square \) Relevance: Multiresolution Analysis\(^\text{TM} \), wavelets.
Refinable functions

Definition. \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k) .
\]

\( (*) \)

“Refinement equation”, classical . . .

□ Relevance: Multiresolution Analysis\(^\text{TM}\), wavelets.

□ “Normally”:
Refinable functions

Definition. \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k). \quad \text{(*)}
\]

"Refinement equation", classical . . .

\[\square\text{ Relevance: Multiresolution Analysis}^\text{TM}, \text{ wavelets.}\]

\[\square\text{ “Normally”: Only the coefficient vector } a \text{ is given explicitly.}\]
Refinable functions

**Definition.** \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi(2 \cdot -k) .
\]

(\*)

"Refinement equation", classical . . .

\square **Relevance:** Multiresolution Analysis\textsuperscript{TM}, wavelets.

\square “Normally”: Only the coefficient vector \( a \) is given explicitly.

\( \triangleright \) \( a \) corresponds to LTI filter (bank)
Refinable functions

Definition. \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi(2 \cdot -k). \tag{*}
\]

"Refinement equation", classical . . .

□ Relevance: Multiresolution Analysis\textsuperscript{TM}, wavelets.

□ "Normally": Only the coefficient vector \( a \) is given explicitly.

▷ \( a \) corresponds to LTI filter (bank) (Linear Time Invariant).
Refinable functions

Definition. \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k).
\]

\( \ast \)

“Refinement equation”, classical . . .

\( \square \) Relevance: Multiresolution Analysis\(^\text{TM} \), wavelets.

\( \square \) “Normally”: Only the coefficient vector \( a \) is given explicitly.

\( \triangleright \) \( a \) corresponds to LTI filter (bank) (Linear Time Invariant).

\( \triangleright \) \( a \) has to be constructed appropriately
Refinable functions

Definition. \( \varphi : \mathbb{R} \rightarrow \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \rightarrow \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi(2 \cdot -k) .
\]

("Refinement equation", classical . . .)

- **Relevance:** Multiresolution Analysis™, wavelets.
- "Normally": Only the coefficient vector \( a \) is given explicitly.
  - \( a \) corresponds to LTI filter (bank) (Linear Time Invariant).
  - \( a \) has to be constructed appropriately – finite support!
Refinable functions

Definition. \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k). \tag{*}
\]

“Refinement equation”, classical . . .

□ Relevance: Multiresolution Analysis\(^{\text{TM}}\), wavelets.

□ “Normally”: Only the coefficient vector \( a \) is given explicitly.

▶ \( a \) corresponds to LTI filter (bank) (Linear Time Invariant).
▶ \( a \) has to be constructed appropriately – finite support!
▶ \( \varphi \) only known implicitly as solution of functional equation (*)

The simple case
Refinable functions

Definition. \( \varphi : \mathbb{R} \to \mathbb{R} \) is called refinable with respect to \( a : \mathbb{Z} \to \mathbb{R} \) if

\[
\varphi = \sum_{j \in \mathbb{Z}} a(k) \varphi (2 \cdot -k) .
\]  

("Refinement equation", classical . . .

- **Relevance**: Multiresolution Analysis\(^\text{TM}\), wavelets.

- "Normally": Only the coefficient vector \( a \) is given explicitly.
  - \( a \) corresponds to LTI filter (bank) (Linear Time Invariant).
  - \( a \) has to be constructed appropriately – finite support!
  - \( \varphi \) only known implicitly as solution of functional equation (*)
  - **Task**: Determine properties of \( \varphi \) from \( a \).
Geometry of refinable functions

Building blocks of refinement equation:
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: $\varphi(2\cdot)$. 
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: $\varphi(2 \cdot )$.

- translates by $k/2$: $\varphi(2 \cdot -k)$.
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: $\varphi(2 \cdot )$.

- translates by $k/2$: $\varphi(2 \cdot -k)$.

- weigthed combination:
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: \( \varphi(2\cdot) \).
- translates by \( k/2 \): \( \varphi(2 \cdot -k) \).
- weighted combination:

\[
\sum_{k \in \mathbb{Z}} \varphi(2 \cdot -k) a(k)
\]
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: \( \varphi(2 \cdot \cdot) \).
- translates by \( k/2 \): \( \varphi(2 \cdot -k) \).
- weighed combination:

\[
\sum_{k \in \mathbb{Z}} \varphi(2 \cdot -k) a(k)
\]
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: \( \varphi(2 \cdot \cdot) \).
- translates by \( k/2 \): \( \varphi(2 \cdot -k) \).
- weighted combination:

\[
\sum_{k \in \mathbb{Z}} \varphi(2 \cdot -k) a(k) =: (\varphi \ast a)(2 \cdot)
\]

“semi–discrete” convolution, order will become important!
Geometry of refinable functions

Building blocks of refinement equation:

- dilated functions: \( \phi(2 \cdot \cdot) \).
- translates by \( k/2 \): \( \phi(2 \cdot -k) \).
- weighed combination:

\[
\sum_{k \in \mathbb{Z}} \phi(2 \cdot -k) a(k) =: (\phi \ast a)(2 \cdot)
\]

“semi–discrete” convolution, order will become important!

- In the example: \( a = (\ldots, 0, \frac{1}{2}, 1, \frac{1}{2}, 0, \ldots) \).
The subdivision operator

☐ Simple computation:
The subdivision operator

- Simple computation:

\[ \varphi \ast c \]
The subdivision operator

Simple computation:

\[ \varphi \ast c = (\varphi \ast a) (2 \cdot) \ast c \]
The subdivision operator

Simple computation:

\[ \varphi * c = (\varphi * a)(2\cdot) * c = \sum_{j \in \mathbb{Z}} \varphi(2 \cdot -j) \sum_{k \in \mathbb{Z}} a(j - 2k) c(k) \]
The subdivision operator

Simple computation:

\[ \varphi \ast c = (\varphi \ast a)(2 \cdot) \ast c = \sum_{j \in \mathbb{Z}} \varphi (2 \cdot - j) \sum_{k \in \mathbb{Z}} a (j - 2k) c(k) \]

\[ =: \varphi \ast S_a c (2 \cdot). \]
The subdivision operator

□ Simple computation:

\[
\phi \ast c = (\phi \ast a) (2 \cdot ) \ast c = \sum_{j \in \mathbb{Z}} \phi (2 \cdot -j) \sum_{k \in \mathbb{Z}} a (j - 2k) c (k)
\]

\[=:\quad \phi \ast S_\alpha c (2 \cdot ).\]

□ The operator

\[
S_\alpha : c \mapsto \sum_{k \in \mathbb{Z}} a (\cdot - 2k) c(k)
\]

is called subdivision operator.
The subdivision operator

Simple computation:

\[ \varphi \ast c = (\varphi \ast a)(2 \cdot) \ast c = \sum_{j \in \mathbb{Z}} \varphi(2 \cdot - j) \sum_{k \in \mathbb{Z}} a(j - 2k) c(k) \]

\[ =: \varphi \ast S_\alpha c(2 \cdot). \]

The operator

\[ S_\alpha : c \mapsto \sum_{k \in \mathbb{Z}} \alpha(\cdot - 2k) c(k) \]

is called subdivision operator.

Alternatively:
The subdivision operator

Simple computation:

\[
\varphi \ast c = (\varphi \ast a)(2 \cdot) \ast c = \sum_{j \in \mathbb{Z}} \varphi(2 \cdot - j) \sum_{k \in \mathbb{Z}} a(j - 2k) c(k) =: \varphi \ast S_a c(2 \cdot).
\]

The operator

\[S_a : c \mapsto \sum_{k \in \mathbb{Z}} a(\cdot - 2k) c(k) \quad (\ldots, 0, c(-1), 0, c(0), 0, c(1), 0, \ldots)\]

is called subdivision operator.

Alternatively: upsampling

The simple case
The subdivision operator

□ Simple computation:

\[
\varphi * c = (\varphi * a)(2 \cdot) * c = \sum_{j \in \mathbb{Z}} \varphi(2 \cdot - j) \sum_{k \in \mathbb{Z}} a(j - 2k) \, c(k)
\]

\[=: \varphi * S_a c (2 \cdot).\]

□ The operator

\[S_a: c \mapsto \sum_{k \in \mathbb{Z}} a(\cdot - 2k) \, c(k) = a(\ldots, 0, c(-1), 0, c(0), 0, c(1), 0, \ldots)\]

is called subdivision operator.

□ Alternatively: upsampling and filtering.
Subdivision and convergence

Because of

\[ \varphi \ast c = \varphi \ast S_\alpha c (2 \cdot) \]
Subdivision and convergence

Because of

\[ \varphi \ast c = \varphi \ast S_a c (2^r) = \cdots = \varphi \ast S_a^r c (2^r) , \quad r \in \mathbb{N}, \]
Subdivision and convergence

Because of

$$\varphi \ast c = \varphi \ast S_a c (2^r \cdot) = \cdots = \varphi \ast S_a^r c (2^r \cdot), \quad r \in \mathbb{N},$$

we have

$$S_a^r c \sim f (2^{-r} \cdot).$$
Subdivision and convergence

Because of

$$\varphi \ast c = \varphi \ast S_a c (2^\cdot) = \cdots = \varphi \ast S_a^r c (2^r \cdot), \quad r \in \mathbb{N},$$

we have

$$S_a^r c \sim f (2^{-r} \cdot).$$

**Definition.** *Subdivision scheme* $S_a$ *is called convergent,*
Subdivision and convergence

Because of

$$\varphi \ast c = \varphi \ast S_a c (2 \cdot) = \cdots = \varphi \ast S_a^r c (2^r \cdot), \quad r \in \mathbb{N},$$

we have

$$S_a^r c \sim f (2^{-r} \cdot).$$

**Definition.** Subdivision scheme $S_a$ is called *convergent*, if for any $c$ there exists a limit function $f$ such that

$$\lim_{r \to \infty} \| S_a^r c - f_c \| = 0$$

and $f_c \neq 0$ for at least one $c$. 


Subdivision and convergence

Because of

\[ \varphi \ast c = \varphi \ast S_a c (2^r) = \cdots = \varphi \ast S_a^r c (2^r), \quad r \in \mathbb{N}, \]

we have

\[ S_a^r c \sim f (2^{-r} \cdot). \]

**Definition.** *Subdivision scheme* $S_a$ *is called convergent*, if for any $c$ there exists a limit function $f$ such that

\[ \lim_{r \to \infty} \| S_a^r c - f_c \| = 0 \]

*Compares apples and peas!*

and $f_c \neq 0$ for at least one $c$. 
Subdivision and convergence

Because of

\[ \varphi \ast c = \varphi \ast S_a c (2^r \cdot) = \cdots = \varphi \ast S_a c (2^r \cdot), \quad r \in \mathbb{N}, \]

we have

\[ S^r_a c \sim f (2^{-r} \cdot). \]

**Definition.** Subdivision scheme \( S_a \) is called convergent, if for any \( c \) there exists a limit function \( f \) such that

\[ \lim_{r \to \infty} \| S^r_a c - f_c \| = 0 \]

Comparer apples and peas! Discrete and continuous funktion!

and \( f_c \neq 0 \) for at least one \( c \).
Subdivision and convergence

Because of

\[ \varphi * c = \varphi * S_a c (2^r) = \cdots = \varphi * S_a c (2^r), \quad r \in \mathbb{N}, \]

we have

\[ S^r_a c \sim f (2^{-r}). \]

**Definition.** Subdivision scheme \( S_a \) is called **convergent**, if for any \( c \) there exists a **limit function** \( f \) such that

\[ \lim_{r \to \infty} \| S^r_a c - f_c \| = 0 \]

*Compares apples and peas! Discrete and continuous funktion!* 

and \( f_c \neq 0 \) for at least one \( c \).

**Remark.** Norm \( \leftrightarrow \) spaces from which \( c \) and \( f_c \) are taken.
Spaces and symbols

- Sequence spaces $\ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z})$, $1 \leq p \leq \infty$. 
Spaces and symbols

- Sequence spaces $\ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z})$, $1 \leq p \leq \infty$.

- Function spaces $H_p = L_p(\mathbb{R})$, $1 \leq p < \infty$, and $H_p = C_u(\mathbb{R})$, $p = \infty$. 
Spaces and symbols

- Sequence spaces $\ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z})$, $1 \leq p \leq \infty$.

- Function spaces $H_p = L_p(\mathbb{R})$, $1 \leq p < \infty$, and $H_p = C_u(\mathbb{R})$, $p = \infty$.

- Operating on discrete lattices $\Rightarrow L_\infty$ makes no sense.
Spaces and symbols

- Sequence spaces \( \ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z}), \ 1 \leq p \leq \infty \).
- Function spaces \( H_p = L_p(\mathbb{R}), \ 1 \leq p < \infty \), and \( H_p = C_u(\mathbb{R}), \ p = \infty \).
- Operating on discrete lattices \( \Rightarrow L_\infty \) makes no sense.

**Definition.** For \( a \in \ell(\mathbb{Z}) \) the Symbol \( a^*(z) \) is the Laurent polynomial

\[
a^*(z) = \sum_{k \in \mathbb{Z}} a(k) z^k, \quad z \in \mathbb{C}_\times := \mathbb{C} \setminus \{0\}.
\]
Spaces and symbols

- Sequence spaces $\ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z})$, $1 \leq p \leq \infty$.
- Function spaces $H_p = L_p(\mathbb{R})$, $1 \leq p < \infty$, and $H_p = C_\text{u}(\mathbb{R})$, $p = \infty$.
- Operating on discrete lattices \(\Rightarrow L_\infty\) makes no sense.

**Definition.** For $a \in \ell(\mathbb{Z})$ the Symbol $a^*(z)$ is the Laurent polynomial

$$a^*(z) = \sum_{k \in \mathbb{Z}} a(k) z^k, \quad z \in \mathbb{C}_x := \mathbb{C} \setminus \{0\}.$$ 

$z^{-1}$ transformation!
Spaces and symbols

- Sequence spaces $\ell_p(\mathbb{Z}) \subset \ell(\mathbb{Z})$, $1 \leq p \leq \infty$.
- Function spaces $H_p = L_p(\mathbb{R})$, $1 \leq p < \infty$, and $H_p = C_u(\mathbb{R})$, $p = \infty$.
- Operating on discrete lattices $\Rightarrow L_\infty$ makes no sense.

**Definition.** For $a \in \ell(\mathbb{Z})$ the Symbol $a^*(z)$ is the Laurent polynomial

$$a^*(z) = \sum_{k \in \mathbb{Z}} a(k) z^k, \quad z \in \mathbb{C}_{\times} := \mathbb{C} \setminus \{0\}.$$

$z^{-1}$ transformation!

**Example.** $(S_\alpha c)^* = a^*(z) c^*(z^2)$. 

The Theorem

Goal:
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.
- can (re)produce polynomials:
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.

- can (re)produce polynomials: $\Pi_k \subset \varphi * \ell(\mathbb{Z})$. 
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.

- can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

  “Approximation order”, vanishing moments of filters, compression rates, . . .
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

☐ is differentiable.

☐ can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

“Approximation order”, vanishing moments of filters, compression rates, . . .

**Theorem.** [classical]
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.
- can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

“Approximation order”, vanishing moments of filters, compression rates, . . .

**Theorem. [classical]**

$\varphi$ has a first derivative if and only if
**The Theorem**

**Goal:** Determine from mask $\alpha$ whether $\varphi$

- is differentiable.
- can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

“Approximation order”, vanishing moments of filters, compression rates, ... 

**Theorem. [classical]**  
$\varphi$ has a first derivative if and only if

1. $\alpha^*(z) = \frac{1}{2} (z + 1) b^*(z)$,
The Theorem

**Goal:** Determine from mask \( \alpha \) whether \( \varphi \)

\( \square \) is differentiable.

\( \square \) can (re)produce polynomials: \( \Pi_k \subset \varphi \ast \ell(\mathbb{Z}) \).

“Approximation order”, vanishing moments of filters, compression rates, \ldots

**Theorem. [classical]**

\( \varphi \) has a first derivative if and only if

1. \( \alpha^*(z) = \frac{1}{2} (z + 1) b^*(z) \),

2. the subdivision scheme \( S_b \) converges.
The Theorem

Goal: Determine from mask \( \alpha \) whether \( \varphi \)

\( \square \) is differentiable.

\( \square \) can (re)produce polynomials: \( \Pi_k \subset \varphi \ast \ell(\mathbb{Z}) \).

“Approximation order”, vanishing moments of filters, compression rates, ...  

Theorem. [classical] \( \varphi \) stable. \( \varphi \) has a first derivative if and only if

1. \( \alpha^*(z) = \frac{1}{2} (z + 1) b^*(z) \),

2. the subdivision scheme \( S_b \) converges.
The Theorem

**Goal:** Determine from mask $a$ whether $\varphi$

- is differentiable.
- can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

“Approximation order”, vanishing moments of filters, compression rates, …

**Theorem. [classical]** $\varphi$ stable.

$\varphi$ has a first derivative if and only if

1. $a^*(z) = \frac{1}{2}(z + 1)b^*(z),$

2. the subdivision scheme $S_b$ converges.

**Stability:** $\|\varphi \ast c\| \sim \|c\|$
The Theorem

**Goal:** Determine from mask $\alpha$ whether $\varphi$

$\square$ is differentiable.

$\square$ can (re)produce polynomials: $\Pi_k \subset \varphi \ast \ell(\mathbb{Z})$.

“Approximation order”, vanishing moments of filters, compression rates, . . .

**Theorem. [classical]** $\varphi$ stable.

$\varphi$ has a first derivative if and only if

1. $a^*(z) = \frac{1}{2} (z + 1) b^*(z)$,

2. the subdivision scheme $S_b$ converges.

**Stability:** $\| \varphi \ast c \| \approx \| c \|$ – equivalence of norms.
Comments

☐ Higher order derivatives by iteration:
Comments

Higher order derivatives by iteration:

\[ \varphi \in H_p^k \iff a^*(z) = \frac{(z + 1)^r}{2^r} b^*(z), \quad S_b, \quad H_p\text{-convergent.} \]
Comments

- Higher order derivatives by iteration:

\[ \varphi \in H^k_p \iff a^*(z) = \frac{(z + 1)^r}{2^r} b^*(z), \quad S_b \quad H_p \text{–convergent.} \]

- Consequence: “smoothing” of known refinable functions.
Comments

Higher order derivatives by iteration:

\[ \varphi \in H_p^k \iff a^*(z) = \frac{(z + 1)^r}{2^r} b^*(z), \quad S_b \quad H_p-\text{convergent}. \]

Consequence: “smoothing” of known refinable functions.
Corresponds to \( \varphi \mapsto \varphi \ast \chi_{[0,1]} \).
Comments

 Higher order derivatives by iteration:

\[ \varphi \in H_p^k \iff a^*(z) = \frac{(z + 1)^r}{2^r} b^*(z), \quad S_b \quad H_p \text{–convergent.} \]

 Consequence: “smoothing” of known refinable functions.
  Corresponds to \( \varphi \mapsto \varphi \ast \chi_{[0,1]} \).

 “Minimal” functions: B–splines.
Comments

- Higher order derivatives by iteration:
  \[ \varphi \in H_p^k \iff a^*(z) = \frac{(z+1)^r}{2^r} b^*(z), \quad S_b \quad H_p \text{-convergent}. \]

- Consequence: “smoothing” of known refinable functions. Corresponds to \( \varphi \mapsto \varphi \ast \chi_{[0,1]} \).

- “Minimal” functions: B–splines.

- Stability is (in general) necessary for characterization.
Comments

□ Higher order derivatives by iteration:

\[ \varphi \in H_k^p \iff \alpha^*(z) = \frac{(z+1)^r}{2^r} b^*(z), \quad S_b \quad H_p – \text{convergent}. \]

□ Consequence: “smoothing” of known refinable functions.
Corresponds to \( \varphi \mapsto \varphi \ast \chi_{[0,1]} \).

□ “Minimal” functions: B–splines.

□ Stability is (in general) necessary for characterization.

□ Stability gives more:
Comments

- Higher order derivatives by iteration:
  \[ \phi \in H_p^k \iff a^*(z) = \frac{(z + 1)^r}{2^r} b^*(z), \quad S_b \quad H_p-\text{convergent}. \]

- Consequence: “smoothing” of known refinable functions. Corresponds to \( \phi \mapsto \phi \ast \chi_{[0,1]} \).

- “Minimal” functions: B–splines.

- Stability is (in general) necessary for characterization.

- Stability gives more:
  \[ \phi \text{ stable and } \alpha-\text{refinable} \Rightarrow S_{\alpha} \text{ converges}. \]
A different interpretation

- Backward difference \( \nabla c := c (\cdot - 1) - c. \)
A different interpretation

- Backward difference $\nabla c := c(z - 1) - c$.
- Satisfies $(\nabla c)^* = (z - 1)c(z)$.
A different interpretation

□ *Backward difference* $\nabla c := c \cdot (\cdot - 1) - c$.

□ Satisfies $(\nabla c)^* = (z - 1) c(z)$.

□ For $a^*(z) = (z + 1) b^*(z)$:
A different interpretation

- **Backward difference** $\nabla c := c (\cdot - 1) - c$.

- Satisfies $(\nabla c)^* = (z - 1) c(z)$.

- For $a^*(z) = (z + 1) b^*(z)$:
  \[ S_a c = a^*(z) c^*(z^2) \]
A different interpretation

- **Backward difference** \( \nabla c := c(z - 1) - c. \)
- Satisfies \((\nabla c)^* = (z - 1)c(z).\)
- For \(a^*(z) = (z + 1)b^*(z):\)

\[
\nabla S_a c = (z - 1) a^*(z) c^* (z^2)
\]
A different interpretation

- **Backward difference** $\nabla c := c(z - 1) - c$.

- Satisfies $(\nabla c)^* = (z - 1) c(z)$.

- For $a^*(z) = (z + 1) b^*(z)$:

  $$\nabla S_a c = (z - 1) a^*(z) c^*(z^2) = b^*(z) (z^2 - 1) c^*(z^2)$$
A different interpretation

- **Backward difference** \( \nabla c := c \cdot -1 - c \).

- Satisfies \((\nabla c)^* = (z - 1) c(z)\).

- For \(a^*(z) = (z + 1) b^*(z)\):

\[
\nabla S_a c = (z - 1) a^*(z) c^*(z^2) = b^*(z) (z^2 - 1) c^*(z^2) = S_b \nabla c
\]
A different interpretation

- Backward difference \( \nabla c := c(\cdot - 1) - c \).

- Satisfies \((\nabla c)^* = (z - 1) c(z)\).

- For \(a^*(z) = (z + 1) b^*(z)\):

\[
\nabla S_a c = (z - 1) a^*(z) c^*(z^2) = b^*(z) (z^2 - 1) c^*(z^2) = S_b \nabla c
\]

Corollary. Stable \( \varphi \) is differentiable if and only if

\[
\nabla S_a = \frac{1}{2} S_b \nabla \quad \text{and} \quad S_b \text{ convergent.}
\]
A different interpretation

- **Backward difference** \( \nabla c := c(z - 1) - c \).

- Satisfies \((\nabla c)^* = (z - 1)c(z)\).

- For \(a^*(z) = (z + 1)b^*(z)\):

  \[
  \nabla S_a c = (z - 1) a^*(z) c^* (z^2) = b^*(z) (z^2 - 1) c^* (z^2) = S_b \nabla c
  \]

**Corollary.** Stable \(\varphi\) is differentiable if and only if

\[
\nabla S_a = \frac{1}{2} S_b \nabla \quad \text{and} \quad S_b \text{ convergent.}
\]

"Factorization" by means of difference operators.
And now to the real stuff

**Ingredients** in $s$ variables:
And now to the real stuff

**Ingredients** in $s$ variables:

- *Dilatation* matrix $\Xi \in \mathbb{Z}^{s \times s}$. 
And now to the real stuff

**Ingredients** in $s$ variables:

- *Dilatation* matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  
  “Expanding”:
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - “Expanding”: eigenvalues $| \cdot | > 1$ or $\|\Xi^{-k}\| \rightarrow 0$
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation matrix** $\Xi \in \mathbb{Z}^{s \times s}$.

  - "Expanding": eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \implies \Xi^{-k}\mathbb{Z}^s$ dense.
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - “Expanding”: eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \Rightarrow \Xi^{-k}\mathbb{Z}^s$ dense.
  - “Isotropic:” alle eigenvalues have the same modulus.
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - "Expanding": eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \Rightarrow \Xi^{-k}\mathbb{Z}^s$ dense.
  - "Isotropic:” alle eigenvalues have the same modulus. $\Rightarrow \kappa(\Xi^k) < C$. 
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation matrix** $\Xi \in \mathbb{Z}^{s \times s}$.
  - "Expanding": eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \Rightarrow \Xi^{-k}\mathbb{Z}^s$ dense.
  - "Isotropic:" alle eigenvalues have the same modulus. $\Rightarrow \kappa(\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$--matrix valued
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation matrix** $\Xi \in \mathbb{Z}^{s \times s}$.
  - **“Expanding”**: eigenvalues $| \cdot | > 1$ or $\| \Xi^{-k} \| \to 0 \Rightarrow \Xi^{-k} \mathbb{Z}^s$ dense.
  - **“Isotropic:”** alle eigenvalues have the same modulus. $\Rightarrow \kappa (\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued

\[ \Phi = \Phi \ast A (\Xi \cdot) \]
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - “Expanding”: eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \Rightarrow \Xi^{-k}\mathbb{Z}^s$ dense.
  - “Isotropic:” alle eigenvalues have the same modulus. $\Rightarrow \kappa (\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued (order!)

  $$\Phi = \Phi \ast A (\Xi)$$
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - “Expanding”: eigenvalues $| \cdot | > 1$ or $\| \Xi^{-k} \| \to 0 \Rightarrow \Xi^{-k} \mathbb{Z}^s$ dense.
  - “Isotropic:” alle eigenvalues have the same modulus. $\Rightarrow \kappa (\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued (order!)
  \[
  \Phi = \Phi \ast A (\Xi \cdot) = \sum_{\alpha \in \mathbb{Z}^s} \Phi (\Xi \cdot - \alpha) A(\alpha), \quad A \in \ell^{\mathbb{N} \times \mathbb{N}} (\mathbb{Z}^s).
  \]
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation matrix** $\Xi \in \mathbb{Z}^{s \times s}$.
  - "Expanding": eigenvalues $| \cdot | > 1$ or $\| \Xi^{-k} \| \to 0 \implies \Xi^{-k} \mathbb{Z}^s$ dense.
  - "Isotropic:" alle eigenvalues have the same modulus. $\implies \kappa (\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued (order!)

\[
\Phi = \Phi \star A (\Xi \cdot) = \sum_{\alpha \in \mathbb{Z}^s} \Phi (\Xi \cdot - \alpha) A(\alpha), \quad A \in \ell^{\mathbb{N} \times \mathbb{N}}(\mathbb{Z}^s).
\]

- Subdivision operator:
And now to the real stuff

Ingredients in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - “Expanding”: eigenvalues $|\cdot| > 1$ or $\|\Xi^{-k}\| \to 0 \implies \Xi^{-k}\mathbb{Z}^s$ dense.
  - “Isotropic:” alle eigenvalues have the same modulus. $\implies \kappa(\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued (order!)
  
  \[
  \Phi = \Phi \ast A(\cdot) = \sum_{\alpha \in \mathbb{Z}^s} \Phi(\Xi \cdot - \alpha) A(\alpha), \quad A \in \ell^{\mathbb{N} \times \mathbb{N}}(\mathbb{Z}^s).
  \]

- Subdivision operator:
  
  \[
  S_A c = \sum_{\alpha \in \mathbb{Z}^s} A(\cdot - \Xi \alpha) c(\alpha), \quad c \in \ell^{\mathbb{N}}(\mathbb{Z}^s).
  \]
And now to the real stuff

**Ingredients** in $s$ variables:

- **Dilatation** matrix $\Xi \in \mathbb{Z}^{s \times s}$.
  - "Expanding": eigenvalues $| \cdot | > 1$ or $\|\Xi^{-k}\| \to 0 \Rightarrow \Xi^{-k}\mathbb{Z}^s$ dense.
  - "Isotropic:" alle eigenvalues have the same modulus. $\Rightarrow \kappa(\Xi^k) < C$.

- Refinable function $\Phi$ is $\mathbb{N} \times \mathbb{N}$–matrix valued (order!)
  \[
  \Phi = \Phi * A(\Xi \cdot) = \sum_{\alpha \in \mathbb{Z}^s} \Phi(\Xi \cdot - \alpha) A(\alpha), \quad A \in \ell^{\mathbb{N} \times \mathbb{N}}(\mathbb{Z}^s).
  \]

- Subdivision operator:
  \[
  S_A c = \sum_{\alpha \in \mathbb{Z}^s} A(\cdot - \Xi \alpha) c(\alpha), \quad c \in \ell^{\mathbb{N} \times K}(\mathbb{Z}^s).
  \]
Why matrix valued?

- Matrix valued masks \(\Rightarrow\) multivariate wavelets!
Why matrix valued?

- Matrix valued masks $\Rightarrow$ *multiwavelets*!
- Simultaneous processing of vector signals:
Why matrix valued?

- Matrix valued masks $\Rightarrow$ multiwavelets!
- Simultaneous processing of vector signals: RGB,
Why matrix valued?

- Matrix valued masks \(\Rightarrow\) *multiwavelets*!
- Simultaneous processing of vector signals: RGB, biosignals – EEG.
Why matrix valued?

- Matrix valued masks ⇒ \textit{multiwavelets}!
- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

The multivariate case – concepts
Why matrix valued?

- Matrix valued masks ⇒ multiwavelets!
- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.
- **Reason:**
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*!
- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.
- **Reason:** Difference operator
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*!
- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.
- **Reason:** Difference operator

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s
\end{bmatrix} \quad c = \begin{bmatrix}
c \cdot + \epsilon_1 - c \\
\vdots \\
c \cdot + \epsilon_s - c
\end{bmatrix}
\]
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*!

- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

- **Reason:** Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s \\
\end{bmatrix}
c = \begin{bmatrix}
c(\cdot + \epsilon_1) - c \\
\vdots \\
c(\cdot + \epsilon_s) - c \\
\end{bmatrix}
\n\]
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*

- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

- **Reason**: Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s
\end{bmatrix}
\]

\[
c = \begin{bmatrix}
(c \cdot + \epsilon_1) - c \\
\vdots \\
(c \cdot + \epsilon_s) - c
\end{bmatrix}
\]

is vector valued.
Why matrix valued?

- Matrix valued masks ⇒ multiwavelets!

- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

- **Reason:** Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\n\nabla_1 \\
\vdots \\
\n\nabla_s \\
\end{bmatrix}
\]

\[
c = \begin{bmatrix}
\c(\cdot + \epsilon_1) - c \\
\vdots \\
\c(\cdot + \epsilon_s) - c \\
\end{bmatrix}
\]

is vector valued.

- **Consequence:**
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*!

- Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

- **Reason:** Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s
\end{bmatrix}
\]

\[
c = \begin{bmatrix}
c (\cdot + \epsilon_1) - c \\
\vdots \\
c (\cdot + \epsilon_s) - c
\end{bmatrix}
\]

is vector valued.

- **Consequence:** \( \nabla S_A = S_B \nabla \)
Why matrix valued?

Matrix valued masks ⇒ multiwavelets!

Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

Reason: Difference operator (discrete gradient)

\[ \nabla c = \begin{bmatrix} \nabla_1 \\ \vdots \\ \nabla_s \end{bmatrix} \quad c = \begin{bmatrix} c(\cdot + \epsilon_1) - c \\ \vdots \\ c(\cdot + \epsilon_s) - c \end{bmatrix} \]

is vector valued.

Consequence: \( \nabla S_A = S_B \nabla \Rightarrow B \in \ell^{Ns \times Ns}(\mathbb{Z}^s) \).
Why matrix valued?

Matrix valued masks ⇒ multiwavelets!

Simultaneous processing of vector signals: RGB, biosignals – EEG. “Multivariate”.

Reason: Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s \\
\end{bmatrix}
c = \begin{bmatrix}
 c (\cdot + \epsilon_1) - c \\
\vdots \\
 c (\cdot + \epsilon_s) - c \\
\end{bmatrix}
\]

is vector valued.

Consequence: \( \nabla S_A = S_B \nabla \) ⇒ \( B \in \mathcal{L}^{N_s \times N_s} (\mathbb{Z}^s) \).

Matrix functions needed for iteration!
Why matrix valued?

- Matrix valued masks ⇒ *multiwavelets*!

- Simultaneous processing of vector signals: RGB, biosignals – EEG. "Multivariate".

- **Reason**: Difference operator (discrete gradient)

\[
\nabla c = \begin{bmatrix}
\nabla_1 \\
\vdots \\
\nabla_s
\end{bmatrix} c = \begin{bmatrix}
c (\cdot + \epsilon_1) - c \\
\vdots \\
c (\cdot + \epsilon_s) - c
\end{bmatrix}
\]

is **vector valued**.

- **Consequence**: \( \nabla S_A = S_B \nabla \) \( \Rightarrow B \in \ell^{Ns \times Ns}(\mathbb{Z}^s) \).

- Matrix functions **needed** for iteration! Higher order differentiability . . .
Factorization – the scalar case

Argument:
Factorization – the scalar case

**Argument:**

- Assumption: $\varphi = \varphi \ast a(\Xi \cdot)$ and $\varphi$ stable.
Factorization – the scalar case

Argument:

□ Assumption: $\varphi = \varphi \ast a (\Xi \cdot)$ and $\varphi$ stable.

□ $\Rightarrow S_\alpha$ converges.
Factorization – the scalar case

Argument:

□ Assumption: \( \varphi = \varphi \star a (\Xi \cdot) \) and \( \varphi \) stable.

□ \( \Rightarrow \) \( S_a \) converges.

□ \( \Rightarrow \) Exists \( B \in \ell^s \times s (\mathbb{Z}^s) \) such that

\[
\nabla S_a = S_B \nabla.
\]
Factorization – the scalar case

Argument:

- Assumption: \( \varphi = \varphi \ast a (\Xi \cdot) \) and \( \varphi \) stable.

- \( \Rightarrow \) \( S_a \) converges.

- \( \Rightarrow \) Exists \( B \in \ell^{s \times s} (\mathbb{Z}^s) \) such that

\[
\nabla S_a = S_B \nabla.
\]

- Can \( a^*(z) \) be factorized?
Factorization – the scalar case

**Argument:**

- Assumption: $\phi = \phi \ast a (\Xi \cdot)$ and $\phi$ stable.
- $\Rightarrow S_\alpha$ converges.
- $\Rightarrow$ Exists $B \in \ell^s \times s (\mathbb{Z}^s)$ such that
  \[ \nabla S_\alpha = S_B \nabla. \]
- Can $a^*(z)$ be factorized?
  - No!
Factorization – the scalar case

Argument:

☐ Assumption: $\varphi = \varphi \ast \alpha(\Xi \cdot)$ and $\varphi$ stable.

☐ $\Rightarrow S_\alpha$ converges.

☐ $\Rightarrow$ Exists $B \in \ell_s \times s(\mathbb{Z}^s)$ such that

$$\nabla S_\alpha = S_B \nabla.$$ 

☐ Can $\alpha^*(z)$ be factorized?

$\triangledown$ No! $\alpha^*(z) = p(z)B^*(z)$ doesn’t even fit in matrix dimensions.
Factorization – the scalar case

Argument:

- Assumption: $\varphi = \varphi \ast a(\Xi \cdot)$ and $\varphi$ stable.

- $\Rightarrow S_a$ converges.

- $\Rightarrow$ Exists $B \in \ell^s \times s(\mathbb{Z}^s)$ such that 

$$\nabla S_a = S_B \nabla.$$ 

- Can $a^*(z)$ be factorized?
  
  - No! $a^*(z) = p(z)B^*(z)$ doesn’t even fit in matrix dimensions.
  
  - Yes!
Factorization – the scalar case

Argument:

□ Assumption: $\varphi = \varphi \ast a(\Xi \cdot)$ and $\varphi$ stable.

□ $\Rightarrow S_\alpha$ converges.

□ $\Rightarrow$ Exists $B \in \ell^s \times s(\mathbb{Z}^s)$ such that

$$\nabla S_\alpha = S_B \nabla.$$

□ Can $a^*(z)$ be factorized?

▷ No! $a^*(z) = p(z)B^*(z)$ doesn’t even fit in matrix dimensions.

▷ Yes! It’s a matter of perspective ...
Factorization – the ideal case

- Laurent polynomials: $\Lambda = \mathbb{R} [z, z^{-1}]$, 
Factorization – the ideal case

- **Laurent polynomials**: \( \Lambda = \mathbb{R}[z, z^{-1}] \),

\[ \Lambda \ni f(z) \]
Factorization – the ideal case

- Laurent polynomials: \( \Lambda = \mathbb{R}[z, z^{-1}] \), finite sum

\[ \Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \]
Factorization – the ideal case

- **Laurent polynomials**: $\Lambda = \mathbb{R}[z, z^{-1}]$, **finite sum**

$$\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_x.$$
Factorization – the ideal case

- Laurent polynomials: \( \Lambda = \mathbb{R} [z, z^{-1}] \), finite sum
  
  \[
  \Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}^s.
  \]

- For \( A = [a_1 \ldots a_s] \) define the Laurent ideal

  \[
  (z^{a_j} - 1)
  \]
Factorization – the ideal case

- **Laurent polynomials**: $\Lambda = \mathbb{R}[z, z^{-1}]$, finite sum

  $$\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_x^s.$$

- For $\Lambda = [a_1 \ldots a_s]$ define the **Laurent ideal**

  $$\sum_{j=1}^{s} q_j(z) (z^{a_j} - 1) : q_j \in \Lambda$$
Factorization – the ideal case

- **Laurent polynomials**: \( \Lambda = \mathbb{R} [z, z^{-1}] \), finite sum

\[
\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_x.
\]

- For \( \Lambda = [a_1 \ldots a_s] \) define the **Laurent ideal**

\[
\left\{ \sum_{j=1}^{s} q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}.
\]
Factorization – the ideal case

□ **Laurent polynomials:** \( \Lambda = \mathbb{R} \left[ z, z^{-1} \right] \), finite sum

\[
\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_{\alpha} z^\alpha, \quad z \in \mathbb{C}_\times.
\]

□ For \( \Lambda = [a_1 \ldots a_s] \) define the **Laurent ideal**

\[
\langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^{s} q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}.
\]
Factorization – the ideal case

- **Laurent polynomials**: \( \Lambda = \mathbb{R} \left[ z, z^{-1} \right] \), finite sum

  \[ \Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_x. \]

- For \( A = [a_1 \ldots a_s] \) define the **Laurent ideal**

  \[ \langle z^A - 1 \rangle := \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^{s} q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}. \]
Factorization – the ideal case

- **Laurent polynomials**: $\Lambda = \mathbb{R}[z, z^{-1}]$, **finite sum**

$$\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_x^s.$$

- For $A = [a_1 \ldots a_s]$ define the **Laurent ideal**

$$\Lambda \supset \langle z^A - 1 \rangle := \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^s q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}.$$
Factorization – the ideal case

- Laurent polynomials: $\Lambda = \mathbb{R}[z, z^{-1}]$, finite sum

$$
\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}^s.
$$

- For $A = [a_1 \ldots a_s]$ define the Laurent ideal

$$
\Lambda \supset \langle z^A - 1 \rangle := \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^{s} q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}.
$$

**Theorem.** $\nabla S_\alpha = S_B \nabla$ if and only if
Factorization – the ideal case

- *Laurent polynomials:* $\Lambda = \mathbb{R} \left[ z, z^{-1} \right]$, **finite** sum

$$\Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}^\times.$$

- For $A = [a_1 \ldots a_s]$ define the *Laurent ideal*

$$\Lambda \supset \langle z^A - 1 \rangle := \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^s q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}.$$

**Theorem.** $\nabla S_\alpha = S_B \nabla$ if and only if

$$a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle$$

*Quotient ideal*
Factorization – the ideal case

■ Laurent polynomials: \( \Lambda = \mathbb{R} \left[ z, z^{-1} \right] \), finite sum

\[ \Lambda \ni f(z) = \sum_{\alpha \in \mathbb{Z}^s} f_\alpha z^\alpha, \quad z \in \mathbb{C}_\times. \]

■ For \( A = [a_1 \ldots a_s] \) define the Laurent ideal

\[ \Lambda \supseteq \langle z^A - 1 \rangle := \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle := \left\{ \sum_{j=1}^s q_j(z) (z^{a_j} - 1) : q_j \in \Lambda \right\}. \]

Theorem. \( \nabla S_a = S_B \nabla \) if and only if

\[ a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \quad \text{and} \quad a^*(z) [z - 1] = B^*(z) [z^\Xi - 1]. \]

Quotient ideal & representation matrix
Quotient ideals

□ \( \mathcal{I}, \mathcal{J} \) Ideals in \( \Lambda \).
Quotient ideals

\( \mathcal{I}, \mathcal{J} \) Ideals in \( \Lambda \).

\( \Box \) Quotient ideal:
Quotient ideals

- $\mathcal{I}, \mathcal{J}$ Ideals in $\Lambda$.

- **Quotient ideal:**
  
  $$\mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \}$$
Quotient ideals

- $\mathcal{I}, \mathcal{J}$ Ideals in $\Lambda$.
- **Quotient ideal:**
  \[ \mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \} \]
- For $s = 1$ we have $\langle z^2 - 1 \rangle : \langle z - 1 \rangle = \langle z + 1 \rangle$. 
Quotient ideals

- \( \mathcal{I}, \mathcal{J} \) Ideals in \( \Lambda \).

- **Quotient ideal:**
  \[ \mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \} \]

- For \( s = 1 \) we have \( \langle z^2 - 1 \rangle : \langle z - 1 \rangle = \langle z + 1 \rangle \).

- Geometrically: \( V(\mathcal{I} : \mathcal{J}) = V(\mathcal{I}) \setminus V(\mathcal{J}) \).
Quotient ideals

- $\mathcal{I}, \mathcal{J}$ Ideals in $\Lambda$.

- **Quotient ideal:**
  \[ \mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \} \]

- For $s = 1$ we have $\langle z^2 - 1 \rangle : \langle z - 1 \rangle = \langle z + 1 \rangle$.

- Geometrically: $V (\mathcal{I} : \mathcal{J}) = V(\mathcal{I}) \setminus V(\mathcal{J})$.

- Here: remove $(1, \ldots, 1)$. 
Quotient ideals

- $\mathcal{I}, \mathcal{J}$ Ideals in $\Lambda$.

- **Quotient ideal:**
  \[ \mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \} \]

- For $s = 1$ we have $\langle z^2 - 1 \rangle : \langle z - 1 \rangle = \langle z + 1 \rangle$.

- Geometrically: $V(\mathcal{I} : \mathcal{J}) = V(\mathcal{I}) \setminus V(\mathcal{J})$.

- Here: remove $(1, \ldots, 1)$.

- **H–Basis** of quotient ideals: masks of *minimal* support.
Quotient ideals

- $\mathcal{I}, \mathcal{J}$ Ideals in $\Lambda$.

- **Quotient ideal:**
  \[ \mathcal{I} : \mathcal{J} := \{ f \in \Lambda : f \cdot \mathcal{J} \subseteq \mathcal{I} \} \]

- For $s = 1$ we have $\langle z^2 - 1 \rangle : \langle z - 1 \rangle = \langle z + 1 \rangle$.

- Geometrically: $V(\mathcal{I} : \mathcal{J}) = V(\mathcal{I}) \setminus V(\mathcal{J})$.

- Here: remove $(1, \ldots, 1)$.

- $H$–Basis of quotient ideals: masks of minimal support.

- $\Lambda$ is no graded Ring, but “Gröbner”–algorithms are possible.
The representation matrix

- Ideal basis in vector form:
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{A} - 1 \rangle \leftrightarrow [z^{A} - 1] \]
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

- Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \)
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

- Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \)
The representation matrix

□ Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

□ Now: \( a^* \in \langle z^ζ - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{ for } j = 1, \ldots, s \)
The representation matrix

- Ideal basis in vector form:

\[
\langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s]
\]

\[A = [a_1 \ldots a_s], \text{ column vectors}\]

- Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{for } j = 1, \ldots, s \)

\[(z_j - 1) \ a^*(z)\]
The representation matrix

□ Ideal basis in vector form:

\[
\langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \iff [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s]
\]

\[A = [a_1 \ldots a_s],\text{ column vectors}\]

□ Now: \(a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{ for } j = 1, \ldots, s\)

\[
\langle z^\Xi - 1 \rangle \ni (z_j - 1) a^*(z)
\]
The representation matrix

□ Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

□ Now: \( a^* \in \langle z^{\Xi} - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{for } j = 1, \ldots, s \)

\[ \langle z^{\Xi} - 1 \rangle \ni (z_j - 1) a^*(z) \]

\( (z^{\xi_k} - 1), \)
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow \begin{bmatrix} z^A - 1 \end{bmatrix} = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{column vectors} \]

- Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{for} j = 1, \ldots, s \)

\[ \langle z^\Xi - 1 \rangle \ni (z_j - 1) a^*(z) = \sum_{k=1}^{s} b_{jk}^*(z) (z^{\xi_k} - 1), \]
The representation matrix

- Ideal basis in vector form:

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

- Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{ for } j = 1, \ldots, s \)

\[ \langle z^\Xi - 1 \rangle \ni (z_j - 1) \ a^*(z) = \sum_{k=1}^{s} b^*_{jk}(z) \ (z^{\xi_k} - 1), \]

\[ a^*(z) \ [z - 1] = B^*(z) \ [z^\Xi - 1], \]
The representation matrix

\[ \langle z^{a_j} - 1 : j = 1, \ldots, s \rangle = \langle z^A - 1 \rangle \leftrightarrow [z^A - 1] = [z^{a_j} - 1 : j = 1, \ldots, s] \]

\[ A = [a_1 \ldots a_s], \text{ column vectors} \]

Now: \( a^* \in \langle z^\Xi - 1 \rangle : \langle z - 1 \rangle \Rightarrow \text{ for } j = 1, \ldots, s \)

\[ \langle z^\Xi - 1 \rangle \ni (z_j - 1) a^*(z) = \sum_{k=1}^{s} b^*_{jk}(z) (z^{\xi_k} - 1), \]

\[ a^*(z) [z - 1] = B^*(z) [z^\Xi - 1], \]

Defines and computes representation matrix \( B^* \).
Factorization – the matrix case

Consider “submasks”

\[ A_{\gamma} := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \]
Factorization – the matrix case

Consider “submasks”

\[ A_{\gamma} := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A(\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1]^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and

\[ A_\gamma y = y, \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \simeq \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and

\[ A_\gamma y = y, \quad \gamma \in \Gamma \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and

\[ y \in \mathbb{R}^n : A_\gamma y = y, \quad \gamma \in \Gamma \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \sim \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and

\[ \mathcal{E}_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \ \gamma \in \Gamma \} \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \simeq \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and

\[ n := \dim \mathbb{E}_A, \quad \mathbb{E}_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \ \gamma \in \Gamma \} \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and invertible matrix \( V \) such that

\[ n := \dim E_A, \quad E_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \ \gamma \in \Gamma \} =: V \mathbb{R}^n. \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and invertible matrix \( V \) such that

\[ n := \dim E_A, \quad E_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \gamma \in \Gamma \} =: V \mathbb{R}^n. \]

Then:
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A(\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \simeq \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and invertible matrix \( V \) such that

\[ n := \dim E_A, \quad E_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \ \gamma \in \Gamma \} =: V \mathbb{R}^n. \]

Then:

\[ \nabla S_A = S_B \nabla, \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \simeq \mathbb{Z}^s / \Xi \mathbb{Z}^s, \]

and invertible matrix \( V \) such that

\[ n := \dim E_A, \quad E_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \ \gamma \in \Gamma \} =: V \mathbb{R}^n. \]

Then:

\[ \nabla S_A = S_B \nabla, \quad \nabla := \nabla_{V,n} \]
Consider “submasks”

\[ A_\gamma := \sum_{\alpha \in \mathbb{Z}^s} A (\gamma - \Xi \alpha), \quad \gamma \in \Gamma := \Xi [0, 1)^s \cap \mathbb{Z}^s \cong \mathbb{Z}^s/\Xi \mathbb{Z}^s, \]

and invertible matrix \( V \) such that

\[ n := \dim \mathbb{E}_A, \quad \mathbb{E}_A := \{ y \in \mathbb{R}^n : A_\gamma y = y, \gamma \in \Gamma \} =: V \mathbb{R}^n. \]

Then:

\[ \nabla S_A = S_B \nabla, \quad \nabla := \nabla_{V,n} := \begin{bmatrix} \nabla_1 I_n & 0 \\ 0 & I_{N-n} \\ \vdots & \vdots \\ \nabla_s I_n & 0 \\ 0 & I_{N-n} \end{bmatrix} V^{-1}. \]
Convergence of $S_B$?

For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on

$$\nabla \ell^{N \times N} (\mathbb{Z}^s) \subset \nabla \ell^{Ns \times N} (\mathbb{Z}^s).$$
Convergence of $S_B$?

\[ \Box \quad \text{For } S_A = \nabla^{-1} S_B \nabla \text{ we only need behavior of } S_B \text{ on } \]

\[ \nabla \ell^{N \times N} \left( \mathbb{Z}^s \right) \subset \nabla \ell^{Ns \times N} \left( \mathbb{Z}^s \right). \]

**Proper** subspace for $s > 1$!
Convergence of $S_B$?

- For $S_A = \nabla^{-1}S_B\nabla$ we only need behavior of $S_B$ on
  
  $$\nabla \ell^{N\times N}(\mathbb{Z}^s) \subset \nabla \ell^{Ns\times N}(\mathbb{Z}^s).$$

  **Proper** subspace for $s > 1$!

- **Example**: $N = 1$ (scalar)
Convergence of $S_B$?

□ For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on

$$\nabla \ell^{N \times N} (\mathbb{Z}^s) \subset \nabla \ell^{Ns \times N} (\mathbb{Z}^s).$$

**Proper** subspace for $s > 1$!

□ **Example:** $N = 1$ (scalar)

$$(\nabla c)^* (z)$$
Convergence of $S_B$?

- For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on

$$\nabla \ell^{N \times N}(\mathbb{Z}^s) \subset \nabla \ell^{Ns \times N}(\mathbb{Z}^s).$$

Proper subspace for $s > 1$!

- Example: $N = 1$ (scalar)

\[
(\nabla c)^*(z) = \begin{bmatrix}
z_1 - 1 \\
\vdots \\
z_s - 1
\end{bmatrix}
\begin{bmatrix}
c^*(z)
\end{bmatrix}
\]
Convergence of $S_B$?

- For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on

$$\nabla \ell^{\mathbb{N} \times \mathbb{N}} (\mathbb{Z}^s) \subset \nabla \ell^{\mathbb{N}_s \times \mathbb{N}} (\mathbb{Z}^s).$$

Proper subspace for $s > 1$!

- Example: $N = 1$ (scalar)

$$(\nabla c)^*(z) = \begin{bmatrix} z_1 - 1 \\ \vdots \\ z_s - 1 \end{bmatrix} c^*(z)$$

Dependency between Components
Convergence of $S_B$?

For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on 

$$\nabla \ell^{N \times N} (\mathbb{Z}^s) \subset \nabla \ell^{Ns \times N} (\mathbb{Z}^s).$$

Proper subspace for $s > 1$!

Example: $N = 1$ (scalar)

$$(\nabla c)^*(z) = \begin{bmatrix} z_1 - 1 \\ \vdots \\ z_s - 1 \end{bmatrix} c^*(z) \Rightarrow q^T (\nabla c)^* = 0, \quad q \in \mathbb{S}(z - 1)$$

Dependency between Components
Convergence of $S_B$?

- For $S_A = \nabla^{-1} S_B \nabla$ we only need behavior of $S_B$ on

\[
\nabla \ell^{N \times N}(Z)^s \subset \nabla \ell^{Ns \times N}(Z)^s.
\]

**Proper** subspace for $s > 1$!

- **Example:** $N = 1$ (scalar)

\[
(\nabla c)^*(z) = \begin{bmatrix}
    z_1 - 1 \\
    \vdots \\
    z_s - 1
\end{bmatrix} 
\]

\[
c^*(z) \quad \Rightarrow \quad q^T (\nabla c)^* = 0, \quad q \in S(z - 1)
\]

Dependency between Components – syzygies
Convergence of $S_B$?

- For $S_A = \nabla^{-1}S_B\nabla$ we only need behavior of $S_B$ on
  $$\nabla \ell^N \times N (\mathbb{Z}^s) \subset \nabla \ell^{N_s} \times N (\mathbb{Z}^s).$$

  Proper subspace for $s > 1$!

- **Example:** $N = 1$ (scalar)

  $$(\nabla c)^*(z) = \begin{bmatrix} z_1 - 1 \\ \vdots \\ z_s - 1 \end{bmatrix} c^*(z) \Rightarrow q^T (\nabla c)^* = 0, \quad q \in \mathbb{S}(z - 1)$$

  Dependency between Components – syzygies

- “Smoothing” is difficult: $B$ has to be of very particular form!
Limited horizon

Theorem. [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 2I$]
$S_A$ converges if and only if
Limited horizon

Theorem. [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 21$]

$S_A$ converges if and only if $\nabla S_A = S_B \nabla$ and
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2 \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\frac{\| S_B^r c \|}{\| c \|}
\]
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 21 \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell_p^N (\mathbb{Z}^s)
\]
Limited horizon

Theorem. [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 2I$]

$S_A$ converges if and only if $\nabla S_A = S_B \nabla$ and

$$\sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r}$$
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2I \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\limsup_{r \to \infty} \sup \left\{ \frac{\| S_B^r c \|}{\| c \|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r}
\]
Limited horizon

**Theorem.** [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \mathcal{E} = 21 \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho(S_B | \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\| S_B^r c \|}{\| c \|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r}
\]
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2I \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho (S_B \mid \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\| S_B^r c \|}{\| c \|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r} \ < \ (\det \Xi)^{1/p}.
\]
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2I \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho (S_B \mid \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.
\]

Restricted spectral radius!
Limited horizon

**Theorem.** [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 2I$]

$S_A$ converges if and only if $\nabla S_A = S_B \nabla$ and

$$\rho (S_B | \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell^N_p (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.$$ 

*Restricted spectral radius!*

□ Restricted convergence: $S_A c \to F_c = \Phi * c$
Limited horizon

**Theorem.** [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 2\mathbb{I}$]

$S_A$ converges if and only if $\nabla S_A = S_B \nabla$ and

$$\rho (S_B | \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell^N_p (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.$$

*Restricted spectral radius!*

□ Restricted *convergence*: $S_A c \to F_c = \Phi \ast c$ for $c \in \nabla \ell_p (\mathbb{Z}^s)$. 

**Restrictions**
Limited horizon

**Theorem.** [Latour, Müller & Nickel: $N = 1$; Charina, Conti & Sauer: $\Xi = 2\mathbb{I}$]

$S_A$ converges if and only if $\nabla S_A = S_B \nabla$ and

$$
\rho \left( S_B \mid \nabla \right) := \limsup_{r \to \infty} \sup \left\{ \frac{\| S_B^r c \|}{\| c \|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r} < (\text{det} \, \Xi)^{1/p}.
$$

*Restricted spectral radius!*

- **Restricted convergence:** $S_A c \to F_c = \Phi \ast c$ for $c \in \nabla \ell_p (\mathbb{Z}^s)$.

- **Restricted stability:** $\| \Phi \ast c \| \simeq \| c \|$
Limited horizon

Theorem. [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2I \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho \left( S_B \mid \nabla \right) := \limsup_{r \to \infty} \sup \left\{ \frac{\| S_B^r c \|}{\| c \|} : c \in \nabla \ell^N_p (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.
\]

Restricted spectral radius!

☐ Restricted convergence: \( S_A c \to F_c = \Phi \ast c \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).

☐ Restricted stability: \( \| \Phi \ast c \| \simeq \| c \| \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).
Limited horizon

Theorem. [Latour, Müller, Nickel: \( N = 1 \); Charina, Conti, Sauer: \( \Xi = 2I \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho (S_B | \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\| S^r_B c \|}{\| c \|} : c \in \nabla \ell^N_p (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.
\]

*Restricted spectral radius!*

\( \square \) Restricted convergence: \( S_A c \to F_c = \Phi * c \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).

\( \square \) Restricted stability: \( \| \Phi * c \| \simeq \| c \| \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).

\( \square \) Restricted refinability:

\[
\Phi = \Phi * A (\Xi).
\]
Limited horizon

**Theorem.** [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 21 \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho \left( S_B \mid \nabla \right) := \limsup_{r \to \infty} \sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell^N_p (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.
\]

*Restricted spectral radius!*

\(\Box\) **Restricted convergence:** \( S_A c \to F_c = \Phi \ast c \) for \( c \in \nabla \ell^p_p (\mathbb{Z}^s) \).

\(\Box\) **Restricted stability:** \( \|\Phi \ast c\| \sim \|c\| \) for \( c \in \nabla \ell^p_p (\mathbb{Z}^s) \).

\(\Box\) **Restricted refinability:**

\[ \Phi = \Phi \ast A (\Xi \cdot) \Leftrightarrow \Phi \ast c = \Phi \ast S_A c (\Xi \cdot) , \]
**Limited horizon**

**Theorem.** [Latour, Müller & Nickel: \( N = 1 \); Charina, Conti & Sauer: \( \Xi = 2 \)]

\( S_A \) converges if and only if \( \nabla S_A = S_B \nabla \) and

\[
\rho (S_B | \nabla) := \limsup_{r \to \infty} \sup \left\{ \frac{\|S_B^r c\|}{\|c\|} : c \in \nabla \ell_p^N (\mathbb{Z}^s) \right\}^{1/r} < (\det \Xi)^{1/p}.
\]

*Restricted spectral radius!*

- **Restricted convergence:** \( S_A c \to F_c = \Phi \ast c \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).
- **Restricted stability:** \( \|\Phi \ast c\| \sim \|c\| \) for \( c \in \nabla \ell_p (\mathbb{Z}^s) \).
- **Restricted refinability:**

\[
\Phi = \Phi \ast \mathbf{A} (\Xi \cdot) \quad \Leftrightarrow \quad \Phi \ast c = \Phi \ast S_A c (\Xi \cdot), \quad c \in \nabla \ell_p (\mathbb{Z}^s).
\]
And what about the constant $2$?

- Reminder:
And what about the constant 2?

□ Reminder: For \( s = N = 1 \):

\[ \varphi \text{ differentiable } \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]
And what about the constant 2?

□ **Reminder:** For \( s = N = 1 \):

\[
\varphi \text{ differentiable } \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.}
\]

□ **How to prove “⇒”:**
And what about the constant 2?

□ **Reminder:** For $s = N = 1$:

\[ \varphi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]

□ How to prove “⇒”:

▷ Set $\psi := \varphi' \nabla^{-1}$. 

Renormalization 19
And what about the constant 2?

□ **Reminder:** For $s = N = 1$:

\[ \varphi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]

□ How to prove “⇒”:

▷ Set $\psi := \varphi' \nabla^{-1}$, i.e. $\psi \nabla = \varphi'$. 
And what about the constant 2?

□ **Reminder:** For $s = N = 1$:

\[ \varphi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]

□ **How to prove “⇒”:**

▷ Set $\psi := \varphi' \nabla^{-1}$, i.e. $\psi \nabla = \varphi'$, hence $\psi \ast \nabla \mathbf{c} = \phi' \ast \mathbf{c}$. 
And what about the constant 2?

□ **Reminder:** For \( s = N = 1 \):

\[ \varphi \text{ differentiable } \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]

□ **How to prove “⇒”:**

- Set \( \psi := \varphi' \nabla^{-1} \), i.e. \( \psi \nabla = \varphi' \), hence \( \psi \ast \nabla c = \phi' \ast c \).
- \( \psi \) is stable.
And what about the constant 2?

□ **Reminder:** For $s = N = 1$:

\[ \varphi \text{ differentiable } \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.} \]

□ **How to prove “⇒”:**

▷ Set $\psi := \varphi' \nabla^{-1}$, i.e. $\psi \nabla = \varphi'$, hence $\psi \ast \nabla c = \phi' \ast c$.
▷ $\psi$ is stable.
▷ $\psi$ is $b$–refinable.
And what about the constant 2?

- **Reminder:** For \( s = N = 1 \):
  \[
  \varphi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.}
  \]

- **How to prove “⇒”:**
  - Set \( \psi := \varphi' \nabla^{-1} \), i.e. \( \psi \nabla = \varphi' \), hence \( \psi \ast \nabla c = \phi' \ast c \).
  - \( \psi \) is stable.
  - \( \psi \) is \( b \)-refinable.
  - \( S_b \) converges.
And what about the constant 2?

□ **Reminder:** For \( s = N = 1 \):

\[
\varphi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.}
\]

□ **How to prove “⇒”:**

- Set \( \psi := \varphi' \nabla^{-1} \), i.e. \( \psi \nabla = \varphi' \), hence \( \psi \ast \nabla c = \varphi' \ast c \).
- \( \psi \) is stable.
- \( \psi \) is \( b \)-refinable.
- \( S_b \) converges.

□ **Factor \( \frac{1}{2} \)** caused by differentiation of

\[
\varphi = \varphi \ast a (2\cdot).
\]
And what about the constant 2?

□ Reminder: For \( s = N = 1 \):

\[
\phi \text{ differentiable} \iff \nabla S_a = \frac{1}{2} S_b \nabla \text{ and } S_b \text{ converges.}
\]

□ How to prove “\( \Rightarrow \)”:

- Set \( \psi := \phi' \nabla^{-1} \), i.e. \( \psi \nabla = \phi' \), hence \( \psi \ast \nabla c = \phi' \ast c \).
- \( \psi \) is stable.
- \( \psi \) is \( b \)-refinable.
- \( S_b \) converges.

□ Factor \( \frac{1}{2} \) caused by differentiation of

\[
\phi = \phi \ast a (2 \cdot).
\]

□ And for general \( \Xi \)?
The general case

\( \Box \quad D = \begin{bmatrix} \frac{\partial}{\partial x_i} : j = 1, \ldots, s \end{bmatrix} \) gradient.
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \] gradient.

\[ \square \] Same procedure:
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \] gradient.

Same procedure: \( \Phi \) stable und \( A \)-refinable relative to \( \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \)
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \square \text{ Same procedure: } \Phi \text{ stable und } A\text{--refinable relative to } \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \]

\[ \text{Set } \Psi \nabla = D \Phi. \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \square \text{ Same procedure: } \Phi \text{ stable und } A\text{–refinable relative to } \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s). \]

\[ \triangleright \text{ Set } \Psi \nabla = D\Phi. \]

\[ \triangleright \Psi \text{ is stable.} \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \square \text{ Same procedure: } \Phi \text{ stable und } A-\text{refinable relative to } \mathcal{L} \subseteq \ell^{N \times N}(\mathbb{Z}^s) \]

\[ \triangleright \text{ Set } \Psi \nabla = D\Phi. \]
\[ \triangleright \Psi \text{ is stable, restricted on } \nabla \mathcal{L}. \]
The general case

\( \square \ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \) gradient.

\( \square \) Same procedure: \( \Phi \) stable und \( A \)-refinable relative to \( \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \)

\( \triangleright \) Set \( \Psi \nabla = D\Phi. \)
\( \triangleright \) \( \Psi \) is stable, restricted on \( \nabla \mathcal{L}. \)
\( \triangleright \) Refinability:
The general case

\[ \nabla = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \text{□ Same procedure: } \Phi \text{ stable und } A-\text{refinable relative to } \mathcal{L} \subset \ell^{N \times N} (\mathbb{Z}^s) \]

\[ \nabla \Psi = D \Phi. \]

\[ \Psi \text{ is stable, restricted on } \nabla \mathcal{L}. \]

\[ \text{Refinability:} \]

\[ \Psi \ast c = \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \text{Same procedure: } \Phi \text{ stable und } A \text{–refinable relative to } \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \]

\[ \triangleright \text{Set } \Psi \nabla = D\Phi. \]
\[ \triangleright \Psi \text{ is stable, restricted on } \nabla \mathcal{L}. \]
\[ \triangleright \text{Refinability:} \]

\[ \Psi \ast c = \Psi \ast S_B c (\Xi) , \quad c \in \nabla \mathcal{L}. \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \text{Same procedure: } \Phi \text{ stable und } A-\text{refinable relative to } \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \]

\[ \triangleright \text{ Set } \Psi \nabla = D \Phi. \]

\[ \triangleright \Psi \text{ is stable, restricted on } \nabla \mathcal{L}. \]

\[ \triangleright \text{ Refinability: } \]

\[ \Psi \ast c = (I_N \otimes \Xi^T) \Psi \ast S_B c (\Xi \cdot), \quad c \in \nabla \mathcal{L}. \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \] gradient.

Same procedure: Φ stable und A–refinable relative to \( \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \)

- Set \( \Psi \nabla = D \Phi \).
- \( \Psi \) is stable, restricted on \( \nabla \mathcal{L} \).
- Refinability:

\[ \Psi * c = (I_N \otimes \Xi^T) \Psi * S_B c (\Xi \cdot), \quad c \in \nabla \mathcal{L}. \]

\( S_B \) converges restricted

\[ \lim_{r \to \infty} S_B^r c = \Phi * c, \quad c \in \nabla \mathcal{L}. \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

\[ \text{Same procedure: } \Phi \text{ stable und } A\text{–refinable relative to } \mathcal{L} \subseteq \ell^{N \times N}(\mathbb{Z}^s) \]

- Set \( \Psi \nabla = D\Phi \).
- \( \Psi \) is stable, restricted on \( \nabla \mathcal{L} \).
- Refinability:
  \[ \Psi \ast c = (I_N \otimes \Xi^T) \Psi \ast S_B c (\Xi \cdot), \quad c \in \nabla \mathcal{L}. \]

- \( S_B \) converges restricted and renormalized (\( X = I_N \otimes \Xi^T \))

\[ \lim_{r \to \infty} X^r S_B^r c = \Phi \ast c, \quad c \in \nabla \mathcal{L}. \]
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{ gradient.} \]

- Same procedure: \( \Phi \) stable und \( A \)-refinable relative to \( \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \)
  - Set \( \Psi \nabla = D\Phi \).
  - \( \Psi \) is stable, restricted on \( \nabla \mathcal{L} \).
  - Refinability:
    \[ \Psi * c = (I_N \otimes \Xi^T) \Psi * S_B c (\Xi \cdot), \quad c \in \nabla \mathcal{L}. \]

- \( S_B \) converges restricted and renormalized \( (X = I_N \otimes \Xi^T) \)
  \[ \lim_{r \to \infty} X^r S_B^r c = \Phi * c, \quad c \in \nabla \mathcal{L}. \]

- **But:**
  - Renormalization 20
The general case

\[ D = \left[ \frac{\partial}{\partial x_j} : j = 1, \ldots, s \right] \text{gradient.} \]

\[ \square \text{Same procedure: } \Phi \text{ stable und } A\text{–refinable relative to } \mathcal{L} \subseteq \ell^{N \times N} (\mathbb{Z}^s) \]

\[ \Rightarrow \text{Set } \Psi \nabla = D\Phi. \]
\[ \Rightarrow \Psi \text{ is stable, restricted on } \nabla \mathcal{L}. \]
\[ \Rightarrow \text{Refinability:} \]
\[ \Psi \ast c = (I_N \otimes \Xi^T) \Psi \ast S_B c (\Xi \cdot), \quad c \in \nabla \mathcal{L}. \]

\[ \Rightarrow S_B \text{ converges restricted and renormalized } (X = I_N \otimes \Xi^T) \]
\[ \lim_{r \to \infty} X^r S_B^r c = \Phi \ast c, \quad c \in \nabla \mathcal{L}. \]

\[ \square \text{But: Convergence is too much!} \]
And one more concept

**Definition.** $S_A$ is called subconvergent with normalization matrix $X$ if
And one more concept

Definition. \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c,
\]
And one more concept

**Definition.** \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}
\]
And one more concept

**Definition.** $S_A$ is called subconvergent with normalization matrix $X$ if

$$\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}$$

*Convergence of subsequences*
And one more concept

**Definition.**  \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}
\]

*Convergence of subsequences*

- \( X \) isotropic
And one more concept

**Definition.** $S_A$ is called subconvergent with normalization matrix $X$ if

$$\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi * c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}$$

*Convergence of subsequences*

$\square$ $X$ isotropic $\Rightarrow$ $W = (\det X)^{-1/s} X$ has only eigenvalues $| \cdot | = 1$. 
And one more concept

**Definition.** $S_A$ is called **subconvergent with** normalization matrix $X$ if

$$
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c,
$$

$r : \mathbb{N} \to \mathbb{N}$ monotonically increasing.

**Convergence of subsequences**

□ $X$ isotropic $\Rightarrow$ $W = (\det X)^{-1/s} X$ has only eigenvalues $| \cdot | = 1$.
$W^r$ contains convergent subsequence.
And one more concept

**Definition.** \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}
\]

*Convergence of subsequences*

- \( X \) isotropic \( \Rightarrow \) \( W = (\det X)^{-1/s} X \) has only eigenvalues \(| \cdot | = 1\).
  \( W^r \) contains convergent subsequence.

- Subconvergence is **not** excentric:
And one more concept

Definition. \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}
\]

Convergence of subsequences

- \( X \) isotropic \( \Rightarrow \) \( W = (\det X)^{-1/s} X \) has only eigenvalues \( | \cdot | = 1 \).
  \( W^r \) contains convergent subsequence.

- Subconvergence is not excentric:

  \( S_a \) convergent \( \Rightarrow \) \( S_{-a} = -S_a \) subconvergent.
And one more concept

Definition. \( S_A \) is called subconvergent with normalization matrix \( X \) if

\[
\lim_{k \to \infty} X^{r(k)} S_B^{r(k)} c = \Phi \ast c, \quad r : \mathbb{N} \to \mathbb{N} \text{ monotonically increasing.}
\]

Convergence of subsequences

- \( X \) isotropic \( \Rightarrow \) \( W = (\det X)^{-1/s} X \) has only eigenvalues \(| \cdot | = 1\). \( W^r \) contains convergent subsequence.

- Subconvergence is not excentric:

\( S_a \) convergent \( \Rightarrow \) \( S_{-a} = -S_a \) subconvergent.

Happens for \( \varphi = \varphi \ast a (-2 \cdot) \).
Subconvergence

- Assumption: $X$ isotropic.
Subconvergence

- **Assumption:** $X$ isotropic.

- Set $\sigma = \rho(X)$. 
Subconvergence

- **Assumption:** $X$ isotropic.
- Set $\sigma = \rho(X)$.
- Subconvergence depends only on $\sigma$!
Subconvergence

□ Assumption: \( X \) isotropic.

□ Set \( \sigma = \rho(X) \).

□ Subconvergence depends only on \( \sigma \! \).

Lemma. \( S_A \) is subconvergent for \( X \) if and only if \( S_{\sigma A} \) is subconvergent.
Subconvergence

- **Assumption**: $X$ isotropic.

- Set $\sigma = \rho(X)$.

- Subconvergence depends only on $\sigma$!

**Lemma.** $S_\mathcal{A}$ is subconvergent for $X$ if and only if $S_{\sigma\mathcal{A}}$ is subconvergent. Different limit functions!
Subconvergence

- **Assumption:** $X$ isotropic.

- Set $\sigma = \rho(X)$.

- Subconvergence depends only on $\sigma$!

**Lemma.** $S_A$ is subconvergent for $X$ if and only if $S_{\sigma A}$ is subconvergent. Different limit functions!

**Lemma. [Factorization]** $S_A$ subconvergent relative to $X$ then there are $B, B'$ such that

$$\nabla X S_A = S_B \nabla \quad \text{resp.} \quad \nabla S_A = S_{B'} \nabla.$$
Differentiability

**Theorem.** $\Phi$ *stable,*
Differentiability

**Theorem.** \( \Phi \) stable,

1. \( \Phi \ast c = X \Phi \ast S \mathcal{A} c (\Xi) , \quad c \in \mathcal{L} , \)
Differentiability

**Theorem.** \( \Phi \) stable,

1. \( \Phi \ast c = X\Phi \ast S_A c (\Xi \cdot) , \quad c \in \mathcal{L} , \)

2. \( n = \dim \mathcal{E}_A \)
Differentiability

**Theorem.** $\Phi$ stable,

1. $\Phi \ast c = X\Phi \ast S_A c (\Xi \cdot), \quad c \in \mathcal{L},$

2. $n = \dim E_A = \dim \mathcal{R} \left( \hat{\Phi}(0) \right)$
Differentiability

**Theorem.** $\Phi$ stable,

1. $\Phi \ast c = X\Phi \ast S_{A}c (\Xi \cdot )$, \quad $c \in \mathcal{L}$,

2. $n = \dim E_{A} = \dim \mathcal{R} (\hat{\Phi}(0)) = \dim \mathcal{R} (\Phi)$. 
Differentiability

**Theorem.** \( \Phi \) stable,

1. \( \Phi \ast c = X\Phi \ast S_{A}c (\Xi \cdot), \quad c \in \mathcal{L}, \)

2. \( n = \dim E_{A} = \dim \mathcal{R} \left( \hat{\Phi}(0) \right) = \dim \mathcal{R} (\Phi). \)

*Additional condition!*
Differentiability

Theorem. \( \Phi \) stable,

1. \( \Phi \ast c = \chi \Phi \ast S_A c (\Xi \cdot), \quad c \in \mathcal{L}, \)

2. \( n = \dim \mathbb{E}_A = \dim \mathcal{R} \left( \hat{\Phi}(0) \right) = \dim \mathcal{R} (\Phi). \)

Additional condition! Convergence for \( 1 < n < N \).
Differentiability

**Theorem.** $\Phi$ stable,

1. $\Phi \ast c = X\Phi \ast S_A c (\Xi \cdot), \quad c \in \mathcal{L},$

2. $n = \dim E_A = \dim \mathcal{R} \left( \hat{\Phi}(0) \right) = \dim \mathcal{R} (\Phi).$

*Additional condition! Convergence for* $1 < n < N.$

$\Phi$ differentiable if and only if there exist $B, V, Y$ such that

$$\nabla V, n S_A = S_B \nabla V, n$$
Differentiability

Theorem. Φ stable,

1. $Φ * c = XΦ * S_A c (Ξ)$, \quad $c \in \mathcal{L}$,

2. $n = \dim E_A = \dim \mathcal{R} \left( \hat{X}(0) \right) = \dim \mathcal{R} (Φ)$.

Additional condition! Convergence for $1 < n < N$.

Φ differentiable if and only if there exist $B, V, Y$ such that

$$\nabla_{V,n} S_A = S_B \nabla_{V,n}$$

and $S_B$ is subconvergent relative to $\nabla_{V,n} \mathcal{L}$ and normalized with Y.
Remarks

\[ Y = P \left( X \otimes \Xi^T \right) P^{-1} \] for permutation \( P \).
Remarks

- \( Y = P (X \otimes \Xi^T) P^{-1} \) for permutation \( P \).
- \( Y \) again isotropic \( \Rightarrow \) iteration.
Remarks

□ \( Y = P \left( X \otimes \Xi^T \right) P^{-1} \) for permutation \( P \).

□ \( Y \) again isotropic \( \Rightarrow \) iteration.

□ All assumptions on \( \Phi \) carry over to \( \Psi = D\Phi \nabla^{-1} \).
Remarks

□ $Y = P (X \otimes \Xi^T) P^{-1}$ for permutation $P$.

□ $Y$ again isotropic $\Rightarrow$ iteration.

□ All assumptions on $\Phi$ carry over to $\Psi = D\Phi \nabla^{-1}$.

□ Criteria for higher order differentiability.
Remarks

- \[ Y = P \left( X \otimes \Xi^T \right) P^{-1} \] for permutation \( P \).

- \( Y \) again isotropic \( \Rightarrow \) iteration.

- All assumptions on \( \Phi \) carry over to \( \Psi = D\Phi \nabla^{-1} \).

- Criteria for higher order differentiability.

- Phenomena and their origin:
Remarks

- $Y = P (X \otimes \Xi^T) P^{-1}$ for permutation $P$.
- $Y$ again isotropic $\Rightarrow$ iteration.
- All assumptions on $\Phi$ carry over to $\Psi = D\Phi \nabla^{-1}$.
- Criteria for higher order differentiability.
- Phenomena and their origin:
  - Matrix masks of increasing size $\leftarrow$ multivariate.
Remarks

- \( \mathbf{Y} = \mathbf{P} (\mathbf{X} \otimes \Xi^T) \mathbf{P}^{-1} \) for permutation \( \mathbf{P} \).

- \( \mathbf{Y} \) again isotropic \( \Rightarrow \) iteration.

- All assumptions on \( \Phi \) carry over to \( \Psi = D\Phi \nabla^{-1} \).

- Criteria for higher order differentiability.

- Phenomena and their origin:
  - Matrix masks of increasing size \( \leftarrow \) multivariate.
  - “Adapted” difference operators \( \leftarrow \) matrix masks.
Remarks

□ $Y = P (X \otimes \Xi^T) P^{-1}$ for permutation $P$.

□ $Y$ again isotropic $\Rightarrow$ iteration.

□ All assumptions on $\Phi$ carry over to $\Psi = D\Phi \nabla^{-1}$.

□ Criteria for higher order differentiability.

□ Phenomena and their origin:
  ▶ Matrix masks of increasing size $\Leftarrow$ multivariate.
  ▶ “Adapted” difference operators $\Leftarrow$ matrix masks.
  ▶ Normalization, subconvergence $\Leftarrow$ dilatation matrix $\Xi$. 
Remarks

□ $Y = P \left( X \otimes \Xi^T \right) P^{-1}$ for permutation $P$.

□ $Y$ again isotropic $\Rightarrow$ iteration.

□ All assumptions on $\Phi$ carry over to $\Psi = D\Phi \nabla^{-1}$.

□ Criteria for higher order differentiability.

□ Phenomena and their origin:
  ▶ Matrix masks of increasing size $\leftarrow$ multivariate.
  ▶ “Adapted” difference operators $\leftarrow$ matrix masks.
  ▶ Normalization, subconvergence $\leftarrow$ dilatation matrix $\Xi$.
  ▶ Restricted convergenz $\leftarrow$ multivariate.
Conclusion

- Univariate → multivariate requires new concepts.
Conclusion

- Univariate $\rightarrow$ multivariate requires new concepts.

- But: they come in naturally!
Conclusion

- Univariate → multivariate requires new concepts.

- But: they come in naturally!

- Hidden when “only” tensor products are considered.
Conclusion

- Univariate $\rightarrow$ multivariate requires new concepts.
- **But:** they come in naturally!
- Hidden when “only” tensor products are considered.
- Difficult: construction of “smooth” wavelets.
Conclusion

- Univariate $\rightarrow$ multivariate requires new concepts.

- But: they come in naturally!

- Hidden when “only” tensor products are considered.

- Difficult: construction of “smooth” wavelets.

- Little known for $s > 2$. 

End of the story
Conclusion

- Univariate → multivariate requires new concepts.

- **But:** they come in naturally!

- Hidden when “only” tensor products are considered.

- Difficult: construction of “smooth” wavelets.

- Little known for $s > 2$. Applications?

End of the story
Conclusion

- Univariate $\rightarrow$ multivariate requires new concepts.

- **But:** they come in naturally!

- Hidden when “only” tensor products are considered.

- Difficult: construction of “smooth” wavelets.

- Little known for $s > 2$. Applications?

- Light at the end of the tunnel?
Univariate $\rightarrow$ multivariate requires new concepts.

But: they come in naturally!

Hidden when “only” tensor products are considered.

Difficult: construction of “smooth” wavelets.

Little known for $s > 2$. Applications?

Light at the end of the tunnel?