Symbolic-Numeric Collaborations (at RICAM and elsewhere)

Christoph Koutschan

Johann Radon Institute for Computational and Applied Mathematics (RICAM)
Austrian Academy of Sciences

October 2, 2015
DK Computational Mathematics
Statusseminar, Strobl

RICAM
JOHANN · RADON · INSTITUTE
FOR COMPUTATIONAL AND APPLIED MATHEMATICS

Doctoral Program
Computational Mathematics
Numerical Analysis and Symbolic Computation

PART I

Symbolic Derivation of Mean-Field PDEs from Lattice-Based Models

(joint work with Helene Ranetbauer, Georg Regensburger, Marie-Therese Wolfram)

Problem

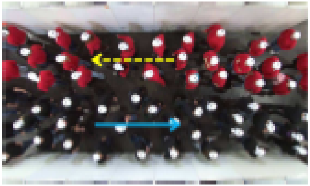
- Simulate crowd motion / pedestrian flow.

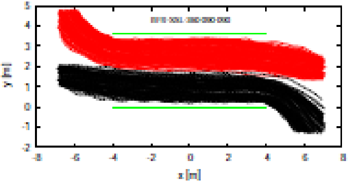
Problem

- Simulate crowd motion / pedestrian flow.
- Consider two types of individuals walking in opposite direction.

Problem

- Simulate crowd motion / pedestrian flow.
- Consider two types of individuals walking in opposite direction.
- Observe lane formation

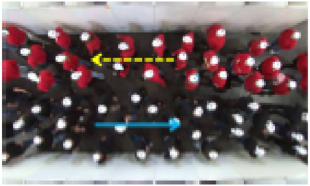


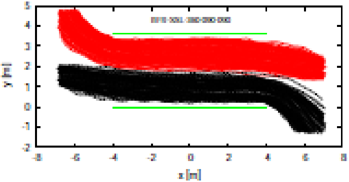


(Experiments done at the research center in Jülich, Germany)

Problem

- Simulate crowd motion / pedestrian flow.
- Consider two types of individuals walking in opposite direction.
- Observe lane formation





(Experiments done at the research center in Jülich, Germany)

Show Mathematica Demo!

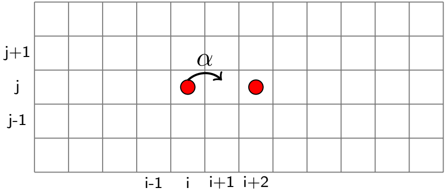
Modelling Assumptions

- Two-dimensional (discrete) lattice
- $r_{i,j}(t) = P(\text{red individual is at position } (i,j) \text{ at time } t)$
- $b_{i,j}(t) = P(\text{blue individual is at position } (i,j) \text{ at time } t)$
- "Density" $\rho_{i,j}(t) = r_{i,j}(t) + b_{i,j}(t)$

Modelling Assumptions

- Two-dimensional (discrete) lattice
- $r_{i,j}(t) = P($red individual is at position (i,j) at time $t)$
- $b_{i,j}(t) = P($blue individual is at position (i,j) at time $t)$
- "Density" $\rho_{i,j}(t) = r_{i,j}(t) + b_{i,j}(t)$
- Probabilities depend on transition rates:

$$T_r^{\{i,j\}\to\{i+1,j\}} = (1-\rho_{i+1,j})(1+\alpha\, r_{i+2,j})$$

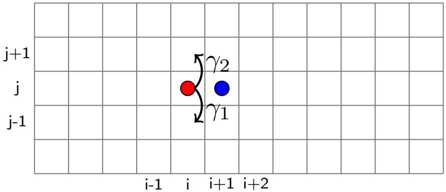


Modelling Assumptions

- Two-dimensional (discrete) lattice
- $r_{i,j}(t) = P(\text{red individual is at position } (i,j) \text{ at time } t)$
- $b_{i,j}(t) = P(\text{blue individual is at position } (i,j) \text{ at time } t)$
- "Density" $\rho_{i,j}(t) = r_{i,j}(t) + b_{i,j}(t)$
- Probabilities depend on transition rates:

$$T_r^{\{i,j\}\to\{i+1,j\}} = (1-\rho_{i+1,j})(1+\alpha\, r_{i+2,j})$$
$$T_r^{\{i,j\}\to\{i,j-1\}} = (1-\rho_{i,j-1})(\gamma_0+\gamma_1\, b_{i+1,j})$$
$$T_r^{\{i,j\}\to\{i,j+1\}} = (1-\rho_{i,j+1})(\gamma_0+\gamma_2\, b_{i+1,j})$$

$$\left(0 \leqslant \alpha \leqslant \tfrac{1}{2}, \quad 0 \leqslant \gamma_0 \leqslant 1, \quad 0 \leqslant \gamma_1, \gamma_2 \leqslant 1\right)$$



Master Equation

The evolution of the system is given by the following equations:

$$\begin{aligned} r_{i,j}(t_{k+1}) &= r_{i,j}(t_k) + \mathcal{T}_r^{\{i-1,j\}\to\{i,j\}} r_{i-1,j}(t_k) \\ &+ \mathcal{T}_r^{\{i,j+1\}\to\{i,j\}} r_{i,j+1}(t_k) + \mathcal{T}_r^{\{i,j-1\}\to\{i,j\}} r_{i,j-1}(t_k) \\ &- \left(\mathcal{T}_r^{\{i,j\}\to\{i+1,j\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j-1\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j+1\}}\right) r_{i,j}(t_k) \end{aligned}$$

$$\begin{aligned} b_{i,j}(t_{k+1}) &= b_{i,j}(t_k) + \mathcal{T}_b^{\{i+1,j\}\to\{i,j\}} b_{i+1,j}(t_k) \\ &+ \mathcal{T}_b^{\{i,j-1\}\to\{i,j\}} b_{i,j-1}(t_k) + \mathcal{T}_b^{\{i,j+1\}\to\{i,j\}} b_{i,j+1}(t_k) \\ &- \left(\mathcal{T}_b^{\{i,j\}\to\{i-1,j\}} + \mathcal{T}_b^{\{i,j\}\to\{i,j+1\}} + \mathcal{T}_b^{\{i,j\}\to\{i,j-1\}}\right) b_{i,j}(t_k) \end{aligned}$$

Master Equation

The evolution of the system is given by the following equations:

$$
\begin{aligned}
r_{i,j}(t_{k+1}) = r_{i,j}(t_k) &+ \mathcal{T}_r^{\{i-1,j\}\to\{i,j\}} r_{i-1,j}(t_k) \\
&+ \mathcal{T}_r^{\{i,j+1\}\to\{i,j\}} r_{i,j+1}(t_k) + \mathcal{T}_r^{\{i,j-1\}\to\{i,j\}} r_{i,j-1}(t_k) \\
&- \left(\mathcal{T}_r^{\{i,j\}\to\{i+1,j\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j-1\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j+1\}}\right) r_{i,j}(t_k)
\end{aligned}
$$

$$\Downarrow$$

$$
\begin{aligned}
r_{i,j}(t_{k+1}) - r_{i,j}(t_k) = &(1 - b_{i,j} - r_{i,j})(1 + \alpha r_{i+1,j}) r_{i-1,j} \\
&+ (\gamma_0 + \gamma_1 b_{i+1,j+1})(1 - b_{i,j} - r_{i,j}) r_{i,j+1} \\
&+ (\gamma_0 + \gamma_2 b_{i+1,j-1})(1 - b_{i,j} - r_{i,j}) r_{i,j-1} \\
&- (1 - b_{i+1,j} - r_{i+1,j})(1 + \alpha r_{i+2,j}) r_{i,j} \\
&- (\gamma_0 + \gamma_1 b_{i+1,j})(1 - b_{i,j-1} - r_{i,j-1}) r_{i,j} \\
&- (\gamma_0 + \gamma_2 b_{i+1,j})(1 - b_{i,j+1} - r_{i,j+1}) r_{i,j}
\end{aligned}
$$

Master Equation

The evolution of the system is given by the following equations:

$$
\begin{aligned}
r_{i,j}(t_{k+1}) &= r_{i,j}(t_k) + \mathcal{T}_r^{\{i-1,j\}\to\{i,j\}} r_{i-1,j}(t_k) \\
&\quad + \mathcal{T}_r^{\{i,j+1\}\to\{i,j\}} r_{i,j+1}(t_k) + \mathcal{T}_r^{\{i,j-1\}\to\{i,j\}} r_{i,j-1}(t_k) \\
&\quad - \left(\mathcal{T}_r^{\{i,j\}\to\{i+1,j\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j-1\}} + \mathcal{T}_r^{\{i,j\}\to\{i,j+1\}}\right) r_{i,j}(t_k)
\end{aligned}
$$

$$\Downarrow$$

$$
\begin{aligned}
r_{i,j}(t_{k+1}) - r_{i,j}(t_k) = {} & (1 - b_{i,j} - r_{i,j})(1 + \alpha r_{i+1,j}) r_{i-1,j} \\
& + (\gamma_0 + \gamma_1 b_{i+1,j+1})(1 - b_{i,j} - r_{i,j}) r_{i,j+1} \\
& + (\gamma_0 + \gamma_2 b_{i+1,j-1})(1 - b_{i,j} - r_{i,j}) r_{i,j-1} \\
& - (1 - b_{i+1,j} - r_{i+1,j})(1 + \alpha r_{i+2,j}) r_{i,j} \\
& - (\gamma_0 + \gamma_1 b_{i+1,j})(1 - b_{i,j-1} - r_{i,j-1}) r_{i,j} \\
& - (\gamma_0 + \gamma_2 b_{i+1,j})(1 - b_{i,j+1} - r_{i,j+1}) r_{i,j}
\end{aligned}
$$

Goal: transform the microscopic equation (recurrence) into a macroscopic equation (differential equation).

Mean-Field PDEs

In an appropriate limit ($\Delta x \to 0, \Delta y \to 0, \Delta t \to 0$), we obtain the following system of PDEs:

$$
\begin{aligned}
\partial_t r = & -\partial_x \left((1-\rho)(1+\alpha r) r\right) + (\gamma_1 - \gamma_2)\partial_y \left((1-\rho) b r\right) \\
& - \tfrac{h}{2}\big(\partial_x^2 (r(1-\rho)(1+\alpha r)) - 2\partial_x((1-\rho)\partial_x r)\big) \\
& + \tfrac{h}{2}\big((\gamma_1+\gamma_2)\partial_y((1-\rho)\partial_y(rb) + br\partial_y \rho) \\
& \quad + 2\gamma_0 \partial_y \left((1-\rho)\partial_y r + r\partial_y \rho\right) \\
& \quad + 2(\gamma_1 - \gamma_2)\partial_y \left((1-\rho) r \partial_x b\right)\big)
\end{aligned}
$$

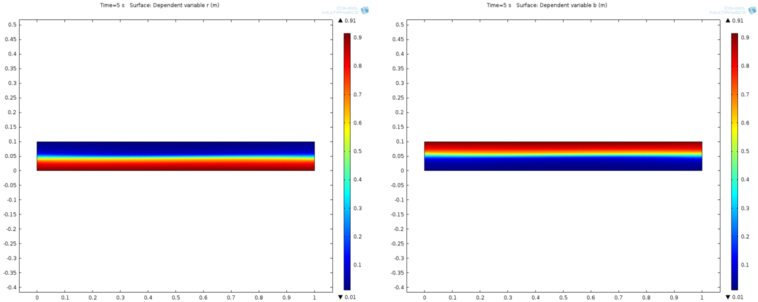
$$
\begin{aligned}
\partial_t b = & \ \partial_x((1-\rho)(1+\alpha b) b) - (\gamma_1 - \gamma_2)\partial_y((1-\rho) b r) \\
& - \tfrac{h}{2}\big(\partial_x^2 (b(1-\rho)(1+\alpha b)) - 2\partial_x((1-\rho)\partial_x b)\big) \\
& + \tfrac{h}{2}\big((\gamma_1+\gamma_2)\partial_y((1-\rho)\partial_y(rb) + br\partial_y \rho) \\
& \quad + 2\gamma_0 \partial_y \left((1-\rho)\partial_y b + b\partial_y \rho\right) \\
& \quad + 2(\gamma_1 - \gamma_2)\partial_y \left((1-\rho) b \partial_x r\right)\big)
\end{aligned}
$$

Parameter settings:

$$\gamma_0 = 0.001, \gamma_1 = 0.6, \gamma_2 = 0.5, \alpha = 0.2, h = 0.1$$

Initial values:

$$r_0 = 0.4 + 0.02 \sin(\pi x) \cos(10\pi y)$$
$$b_0 = 0.4 - 0.02 \sin(\pi x) \cos(10\pi y)$$



(a) red individuals

(b) blue individuals

Taylor Expansion

The transition from discrete to continuous is done by formal Taylor expansions

$$\begin{aligned} r_{i+1,j} &= r_{i,j} + h\partial_x r_{i,j} + \tfrac{1}{2}h^2\partial_x^2 r_{i,j} + \mathcal{O}(h^3) \\ &\vdots \\ b_{i,j+1} &= b_{i,j} + h\partial_y b_{i,j} + \tfrac{1}{2}h^2\partial_y^2 b_{i,j} + \mathcal{O}(h^3) \end{aligned}$$

which are plugged into the master equations

$$\begin{aligned} r_{i,j}(t_{k+1}) - r_{i,j}(t_k) = {} & (1 - b_{i,j} - r_{i,j})(1 + \alpha r_{i+1,j}) r_{i-1,j} \\ & + (\gamma_0 + \gamma_1 b_{i+1,j+1})(1 - b_{i,j} - r_{i,j}) r_{i,j+1} \\ & + (\gamma_0 + \gamma_2 b_{i+1,j-1})(1 - b_{i,j} - r_{i,j}) r_{i,j-1} \\ & - (1 - b_{i+1,j} - r_{i+1,j})(1 + \alpha r_{i+2,j}) r_{i,j} \\ & - (\gamma_0 + \gamma_1 b_{i+1,j})(1 - b_{i,j-1} - r_{i,j-1}) r_{i,j} \\ & - (\gamma_0 + \gamma_2 b_{i+1,j})(1 - b_{i,j+1} - r_{i,j+1}) r_{i,j} \end{aligned}$$

$\longrightarrow$ Such a task calls for Symbolic Computation!

Symbolic Computation

We have experienced:

- Expanded form consists of 173 terms.

Symbolic Computation

We have experienced:

- Expanded form consists of 173 terms.
- Polynomial reduction modulo h^2 results in 62 terms.

Symbolic Computation

We have experienced:

- Expanded form consists of 173 terms.
- Polynomial reduction modulo h^2 results in 62 terms.
- Further simplification is necessary...

Symbolic Computation

We have experienced:

- Expanded form consists of 173 terms.
- Polynomial reduction modulo h^2 results in 62 terms.
- Further simplification is necessary...

For the expansion, we implemented a very general procedure:

- expansion order can be chosen arbitrarily
- number of expansion variables is not fixed
- allow discrete steps of any size: $r_{i+a,j+b}$ for $a, b \in \mathbb{Z}$

Simplification

Goal: write the PDE in a "conservative formulation":

$$
\begin{aligned}
\partial_t r = & -\partial_x \left((1-\rho)(1+\alpha r)r\right) + (\gamma_1-\gamma_2)\partial_y \left((1-\rho)br\right) \\
& - \tfrac{h}{2}\big(\partial_x^2(r(1-\rho)(1+\alpha r)) - 2\partial_x((1-\rho)\partial_x r)\big) \\
& + \tfrac{h}{2}\big((\gamma_1+\gamma_2)\partial_y((1-\rho)\partial_y(rb) + br\partial_y\rho) \\
& \qquad + 2\gamma_0\partial_y\left((1-\rho)\partial_y r + r\partial_y\rho\right) \\
& \qquad + 2(\gamma_1-\gamma_2)\partial_y\left((1-\rho)r\partial_x b\right)\big)
\end{aligned}
$$

Simplification

Goal: write the PDE in a "conservative formulation":

$$\begin{aligned}\partial_t r = &-\partial_x\left((1-\rho)(1+\alpha r)r\right) + (\gamma_1-\gamma_2)\partial_y\left((1-\rho)br\right)\\ &- \tfrac{h}{2}\big(\partial_x^2(r(1-\rho)(1+\alpha r)) - 2\partial_x((1-\rho)\partial_x r)\big)\\ &+ \tfrac{h}{2}\big((\gamma_1+\gamma_2)\partial_y((1-\rho)\partial_y(rb) + br\partial_y\rho)\\ &\qquad + 2\gamma_0\partial_y\left((1-\rho)\partial_y r + r\partial_y\rho\right)\\ &\qquad + 2(\gamma_1-\gamma_2)\partial_y\left((1-\rho)r\partial_x b\right)\big)\end{aligned}$$

This means simplifications of the form $(\partial_x r)b + r(\partial_x b) \rightsquigarrow \partial_x(rb)$.

Simplification

Goal: write the PDE in a "conservative formulation":

$$\begin{aligned}\partial_t r = &-\partial_x\left((1-\rho)(1+\alpha r)r\right) + (\gamma_1-\gamma_2)\partial_y\left((1-\rho)br\right)\\ &-\tfrac{h}{2}\big(\partial_x^2(r(1-\rho)(1+\alpha r)) - 2\partial_x((1-\rho)\partial_x r)\big)\\ &+\tfrac{h}{2}\big((\gamma_1+\gamma_2)\partial_y((1-\rho)\partial_y(rb) + br\partial_y\rho)\\ &\qquad + 2\gamma_0\partial_y\left((1-\rho)\partial_y r + r\partial_y\rho\right)\\ &\qquad + 2(\gamma_1-\gamma_2)\partial_y\left((1-\rho)r\partial_x b\right)\big)\end{aligned}$$

This means simplifications of the form $(\partial_x r)b + r(\partial_x b) \rightsquigarrow \partial_x(rb)$.

Hence we need to perform symbolic integration

- with remainder
- with several unspecified functions
- in several variables.

Symbolic Integration

- Classically, one wants to integrate a concrete function, e.g.,

$$\int \frac{(x+1)^2 + (3x+1)\sqrt{x+\log x}}{x\sqrt{x+\log x}(x+\sqrt{x+\log x})}\,\mathrm{d}x = 2\big(\sqrt{x+\log x} + \log\big(x+\sqrt{x+\log x}\big)\big).$$

Symbolic Integration

- Classically, one wants to integrate a concrete function, e.g.,

$$\int \frac{(x+1)^2 + (3x+1)\sqrt{x+\log x}}{x\sqrt{x+\log x}(x+\sqrt{x+\log x})}\,\mathrm{d}x = \\ 2\left(\sqrt{x+\log x} + \log\left(x+\sqrt{x+\log x}\right)\right).$$

- Now an expression with unspecified functions: both Mathematica and Maple correctly compute

$$\int \left(f^2(\partial_x^2 g) - 2(\partial_x f)^2 g - 2f(\partial_x^2 f)g\right)\mathrm{d}x = f^2(\partial_x g) - 2f(\partial_x f)g.$$

Symbolic Integration

- Classically, one wants to integrate a concrete function, e.g.,

$$\int \frac{(x+1)^2 + (3x+1)\sqrt{x+\log x}}{x\sqrt{x+\log x}(x+\sqrt{x+\log x})}\,\mathrm{d}x = 2\bigl(\sqrt{x+\log x} + \log\bigl(x+\sqrt{x+\log x}\bigr)\bigr).$$

- Now an expression with unspecified functions: both Mathematica and Maple correctly compute

$$\int \bigl(f^2(\partial_x^2 g) - 2(\partial_x f)^2 g - 2f(\partial_x^2 f)g\bigr)\,\mathrm{d}x = f^2(\partial_x g) - 2f(\partial_x f)g.$$

- It is not straightforward to obtain a decomposition into an integrable part and a remainder using the standard integration commands provided by the computer algebra system, e.g.,

$$f\cdot\partial_x f + f = \partial_x\bigl(\tfrac{1}{2}f^2\bigr) + f.$$

Symbolic Integration

- Classically, one wants to integrate a concrete function, e.g.,

$$\int \frac{(x+1)^2 + (3x+1)\sqrt{x+\log x}}{x\sqrt{x+\log x}(x+\sqrt{x+\log x})}\,\mathrm{d}x = 2\big(\sqrt{x+\log x} + \log\big(x+\sqrt{x+\log x}\big)\big).$$

- Now an expression with unspecified functions: both Mathematica and Maple correctly compute

$$\int \big(f^2(\partial_x^2 g) - 2(\partial_x f)^2 g - 2f(\partial_x^2 f)g\big)\,\mathrm{d}x = f^2(\partial_x g) - 2f(\partial_x f)g.$$

- It is not straightforward to obtain a decomposition into an integrable part and a remainder using the standard integration commands provided by the computer algebra system, e.g.,

$$f \cdot \partial_x f + f = \partial_x\big(\tfrac{1}{2}f^2\big) + f.$$

- Multivariate setting: we have several unspecified functions and several variables with respect to which we differentiate.

Algorithmic Idea

Let E be a polynomial expression in a single unspecified function f and its derivatives $\partial_x f, \partial_x^2 f, \ldots, \partial_x^n f$.

Algorithmic Idea

Let E be a polynomial expression in a single unspecified function f and its derivatives $\partial_x f, \partial_x^2 f, \ldots, \partial_x^n f$.

- If E is the derivative of some other polynomial, then the highest derivative occurs linearly, e.g.,

$$\partial_x\big((\partial_x^i f)^k\big) = k(\partial_x^i f)^{k-1}(\partial_x^{i+1} f).$$

Algorithmic Idea

Let E be a polynomial expression in a single unspecified function f and its derivatives $\partial_x f, \partial_x^2 f, \ldots, \partial_x^n f$.

- If E is the derivative of some other polynomial, then the highest derivative occurs linearly, e.g.,

$$\partial_x\big((\partial_x^i f)^k\big) = k(\partial_x^i f)^{k-1}(\partial_x^{i+1} f).$$

- Hence all terms that are nonlinear in $\partial_x^n f$ go to the remainder.

Algorithmic Idea

Let E be a polynomial expression in a single unspecified function f and its derivatives $\partial_x f, \partial_x^2 f, \ldots, \partial_x^n f$.

- If E is the derivative of some other polynomial, then the highest derivative occurs linearly, e.g.,

$$\partial_x\big((\partial_x^i f)^k\big) = k(\partial_x^i f)^{k-1}(\partial_x^{i+1} f).$$

- Hence all terms that are nonlinear in $\partial_x^n f$ go to the remainder.
- Consider a term of the form $u \cdot (\partial_x^{n-1} f)^m(\partial_x^n f)$, where m is the highest power of $\partial_x^{n-1} f$. Then integration by parts yields

$$u \cdot \big(\partial_x^{n-1} f\big)^m(\partial_x^n f) = \partial_x\Big(\frac{u}{m+1}\big(\partial_x^{n-1} f\big)^{m+1}\Big) - \frac{\partial_x u}{m+1}\big(\partial_x^{n-1} f\big)^{m+1}.$$

Ambiguities

The result is not unique for several reasons:

- Several unspecified functions are involved:

$$\begin{aligned}(\partial_x f)(\partial_x g) &= \partial_x\big(f(\partial_x g)\big) - f(\partial_x^2 g)\\ &= \partial_x\big((\partial_x f)g\big) - (\partial_x^2 f)g.\end{aligned}$$

Ambiguities

The result is not unique for several reasons:

- Several unspecified functions are involved:

$$\begin{aligned}(\partial_x f)(\partial_x g) &= \partial_x\big(f(\partial_x g)\big) - f(\partial_x^2 g)\\ &= \partial_x\big((\partial_x f)g\big) - (\partial_x^2 f)g.\end{aligned}$$

- The unspecified function depends on several variables:

$$\begin{aligned}&(\partial_x f)(\partial_y f) + \partial_x f + \partial_y f\\ &\qquad = \partial_x\big(f \cdot \partial_y f + f\big) + \partial_y\big(f\big) - f \cdot \partial_x \partial_y f\\ &\qquad = \partial_x\big(f\big) + \partial_y\big(f \cdot \partial_x f + f\big) - f \cdot \partial_x \partial_y f.\end{aligned}$$

Algorithm

Given:

- E: differential polynomial expression
- $f_1, \ldots, f_k$: unspecified functions
- $x, y, \ldots, z$: integration variables
- d: the desired maximal integration depth of the output:

$$\partial_x(\partial_x(\cdots(\partial_x(I) + R_d)\cdots) + R_1) + R_0.$$

Algorithm

Given:

- E: differential polynomial expression
- $f_1, \ldots, f_k$: unspecified functions
- $x, y, \ldots, z$: integration variables
- d: the desired maximal integration depth of the output:

$$\partial_x(\partial_x(\cdots(\partial_x(I) + R_d)\cdots) + R_1) + R_0.$$

Treat the variables in a fixed order as the main loop.
For each integration variable, say x, proceed as follows:

- Determine the highest derivative with respect to x, say $\partial_x^n f_i$ for some $1 \leqslant i \leqslant k$.
- For each f_i (in the order as specified by the user) the terms involving $\partial_x^n f_i$ are treated.
- Derivatives of order $n+1$ can be produced $\to$ keep these terms.
- We continue by considering derivatives of order $n-1$.

Conclusion and Outlook

Conclusion:

- When we apply our Mathematica implementation of this algorithm to the example from above, we obtain basically the manually derived equation.
- The algorithm can be applied to a more general class of models.
- Paper and software can be found on:

  http://www.koutschan.de/data/meanfield/

Conclusion and Outlook

Conclusion:

- When we apply our Mathematica implementation of this algorithm to the example from above, we obtain basically the manually derived equation.
- The algorithm can be applied to a more general class of models.
- Paper and software can be found on:

  http://www.koutschan.de/data/meanfield/

Outlook:

- Include more features: diagonal stepping, pushing other individuals, . . .
- A symbolic computation approach would be useful to find an appropriate entropy functional for a system.

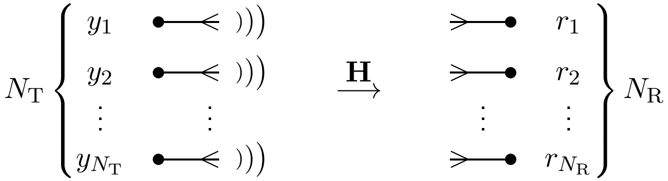
PART II

Computer-Algebra-Based MIMO Performance Analysis

(joint work with Constantin Siriteanu, Akimichi Takemura, Satoshi Kuriki, Donald St. P. Richards, Hyundong Shin)

MIMO Wireless Communication Systems

MIMO = Multiple Input + Multiple Output:



Notation:

- N_T: number of transmitting antennas
- N_R: number of receiving antennas
- $\mathbf{y} = (y_1, y_2, \ldots, y_{N_\mathrm{T}})^\mathcal{T} \in \mathbb{C}^{N_\mathrm{T}}$: transmitted signal vector
- $\mathbf{H}$: the $N_\mathrm{R} \times N_\mathrm{T}$ channel matrix
- $\mathbf{r} = (r_1, r_2, \ldots, r_{N_\mathrm{R}})^\mathcal{T} = \mathbf{Hy} + \mathbf{n}$: received signal vector, where $\mathbf{n}$ is some additive zero-mean Gaussian noise

Channel Matrix

The channel matrix is modeled as a complex-valued Gaussian random matrix, written as

$$\mathbf{H} = \mathbf{H}_{\mathrm{d}} + \mathbf{H}_{\mathrm{r}}$$

where $\mathbf{H}_{\mathrm{d}}$ denotes the deterministic component ("mean") and $\mathbf{H}_{\mathrm{r}}$ the random component.

Channel Matrix

The channel matrix is modeled as a complex-valued Gaussian random matrix, written as

$$\mathbf{H} = \mathbf{H}_{\mathrm{d}} + \mathbf{H}_{\mathrm{r}}$$

where $\mathbf{H}_{\mathrm{d}}$ denotes the deterministic component ("mean") and $\mathbf{H}_{\mathrm{r}}$ the random component.

Fading:

- Rayleigh fading, i.e., $\mathbf{H}_{\mathrm{d}} = 0$ (previous work)
- Rician fading, i.e., $\mathbf{H}_{\mathrm{d}} \neq 0$ (current work)

Channel Matrix

The channel matrix is modeled as a complex-valued Gaussian random matrix, written as

$$\mathbf{H} = \mathbf{H}_{\mathrm{d}} + \mathbf{H}_{\mathrm{r}}$$

where $\mathbf{H}_{\mathrm{d}}$ denotes the deterministic component ("mean") and $\mathbf{H}_{\mathrm{r}}$ the random component.

Fading:

- Rayleigh fading, i.e., $\mathbf{H}_{\mathrm{d}} = 0$ (previous work)
- Rician fading, i.e., $\mathbf{H}_{\mathrm{d}} \neq 0$ (current work)

For sake of simplicity (not w.l.o.g.!), certain assumptions on $\mathbf{H}$:

- $\mathbf{H}_{\mathrm{d}}$ has rank 1
- further assumptions (zero row correlation, etc.)

Zero-Forcing Detection

Recall:

$$\mathbf{r} = \mathbf{H}\mathbf{y} + \mathbf{n}.$$

Zero-Forcing means finding the (modulation constellation) symbols closest to each element of the vector

$$\left(\mathbf{H}^{\mathcal{H}}\mathbf{H}\right)^{-1}\mathbf{H}^{\mathcal{H}}\mathbf{r} = \mathbf{y} + \left(\mathbf{H}^{\mathcal{H}}\mathbf{H}\right)^{-1}\mathbf{H}^{\mathcal{H}}\mathbf{n}.$$

Goal of the analysis: say something about the quality of the connection, i.e., how many symbols are transmitted correctly in average.

The following parameters will be used:

- $N = N_{\mathrm{R}} - N_{\mathrm{T}} + 1$
- x_1, x_2: related to $\|\mathbf{H}_{\mathrm{d}}\|^2 / \mathbb{E}\{\|\mathbf{H}_{\mathrm{r}}\|^2\}$
- Γ_1: related to the additive noise

Signal-to-Noise Ratio (SNR)

The SNR is the ultimate performance measure (determines the quality of the connection).

Theorem. The moment generating function $M(s; x_1, x_2)$ of the SNR for zero-forcing under full-Rician fading with $r = 1$ is

$$M(s; x_1, x_2) = \frac{e^{-x_2}}{(1 - \Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s x_1}{1 - \Gamma_1 s}\right).$$

Signal-to-Noise Ratio (SNR)

The SNR is the ultimate performance measure (determines the quality of the connection).

Theorem. The moment generating function $M(s; x_1, x_2)$ of the SNR for zero-forcing under full-Rician fading with $r = 1$ is

$$M(s; x_1, x_2) = \frac{e^{-x_2}}{(1 - \Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s x_1}{1 - \Gamma_1 s}\right).$$

Definition. The **hypergeometric function** ${}_1F_1$ is defined by

$${}_1F_1(a; b; z) := \sum_{k=0}^{\infty} \frac{(a)_k}{(b)_k} \frac{z^k}{k!}, \quad \text{where}$$

$$(a)_k := a \cdot (a+1) \cdots (a+k-1), \quad (a)_0 := 1$$

is the **Pochhammer symbol** (or rising factorial).

Signal-to-Noise Ratio (SNR)

The SNR is the ultimate performance measure (determines the quality of the connection).

Theorem. The moment generating function $M(s; x_1, x_2)$ of the SNR for zero-forcing under full-Rician fading with $r = 1$ is

$$M(s; x_1, x_2) = \frac{e^{-x_2}}{(1-\Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s x_1}{1-\Gamma_1 s}\right) =$$

$$e^{-x_2} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(N)_{n_1}}{(n_2+N_{\mathrm{R}})_{n_1}} \frac{x_1^{n_1}}{n_1!} \frac{x_2^{n_2}}{n_2!} \sum_{m_1=0}^{n_1} \binom{n_1}{m_1} \frac{(-1)^{m_1}}{(1-s\Gamma_1)^{N+n_1-m_1}}.$$

Signal-to-Noise Ratio (SNR)

The SNR is the ultimate performance measure (determines the quality of the connection).

Theorem. The moment generating function $M(s; x_1, x_2)$ of the SNR for zero-forcing under full-Rician fading with $r = 1$ is

$$M(s; x_1, x_2) = \frac{e^{-x_2}}{(1-\Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s x_1}{1-\Gamma_1 s}\right) =$$

$$e^{-x_2} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(N)_{n_1}}{(n_2 + N_{\mathrm{R}})_{n_1}} \frac{x_1^{n_1}}{n_1!} \frac{x_2^{n_2}}{n_2!} \sum_{m_1=0}^{n_1} \binom{n_1}{m_1} \frac{(-1)^{m_1}}{(1-s\Gamma_1)^{N+n_1-m_1}}.$$

Obtain the SNR probability density function by Laplace transform:

$$\frac{1}{(1-s\Gamma_1)^{N+n_1-m_1}} \quad \overset{\text{Laplace}}{\longleftrightarrow} \quad \frac{t^{N+n_1-m_1-1} e^{-t/\Gamma_1}}{(N+n_1-m_1-1)! \, \Gamma_1^{N+n_1-m_1}}$$

SNR Probability Density Function

Thus we obtain for the SNR probability density function $p(t; x_1, x_2)$:

$$\begin{aligned} p(t; x_1, x_2) &= \int_0^\infty e^{-st} M(s; x_1, x_2)\, \mathrm{d}s \\ &= e^{-x_2} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(N)_{n_1}}{(n_2 + N_{\mathrm{R}})_{n_1}} \frac{x_1^{n_1}}{n_1!} \frac{x_2^{n_2}}{n_2!} \\ &\quad \times \sum_{m_1=0}^{n_1} \binom{n_1}{m_1} \frac{(-1)^{m_1} t^{N+n_1-m_1-1} e^{-t/\Gamma_1}}{(N + n_1 - m_1 - 1)!\, \Gamma_1^{N+n_1-m_1}}. \end{aligned}$$

SNR Probability Density Function

Thus we obtain for the SNR probability density function $p(t; x_1, x_2)$:

$$p(t; x_1, x_2) = \int_0^\infty e^{-st} M(s; x_1, x_2)\,\mathrm{d}s$$
$$= e^{-x_2} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(N)_{n_1}}{(n_2 + N_\mathrm{R})_{n_1}} \frac{x_1^{n_1}}{n_1!} \frac{x_2^{n_2}}{n_2!}$$
$$\times \sum_{m_1=0}^{n_1} \binom{n_1}{m_1} \frac{(-1)^{m_1} t^{N+n_1-m_1-1} e^{-t/\Gamma_1}}{(N + n_1 - m_1 - 1)!\, \Gamma_1^{N+n_1-m_1}}.$$

Definition. Using this, we define the main object of interest, the **outage probability** $P_\mathrm{o}(x_1, x_2)$:

$$P_\mathrm{o}(x_1, x_2) = \int_0^\tau p(t; x_1, x_2)\,\mathrm{d}t$$

where τ is a certain prescribed SNR threshold.

Evaluate

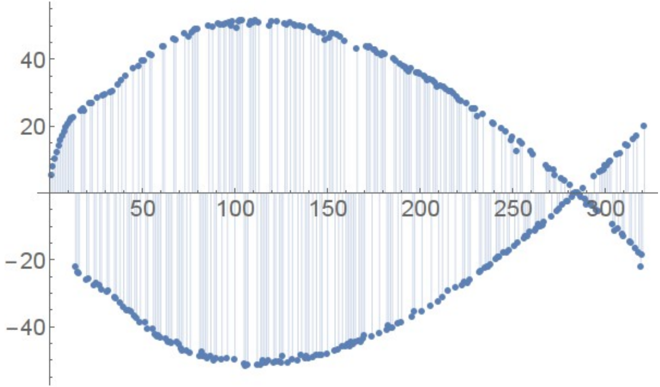
Now, for certain choices of the parameters $N_{\mathrm{R}}, N, x_1, x_2, \Gamma_1, \tau$, we want to "compute" (i.e., evaluate numerically) the outage probability.

First try: truncate the infinite series

$$P_{\mathrm{o}}(x_1, x_2) = e^{-x_2} \sum_{n_1=0}^{\infty} \sum_{n_2=0}^{\infty} \frac{(N)_{n_1}}{(n_2 + N_{\mathrm{R}})_{n_1}} \frac{x_1^{n_1}}{n_1!} \frac{x_2^{n_2}}{n_2!} \times \sum_{m_1=0}^{n_1} \binom{n_1}{m_1} \frac{(-1)^{m_1} \gamma(N + n_1 - m_1, \tau/\Gamma_1)}{(N + n_1 - m_1 - 1)!}$$

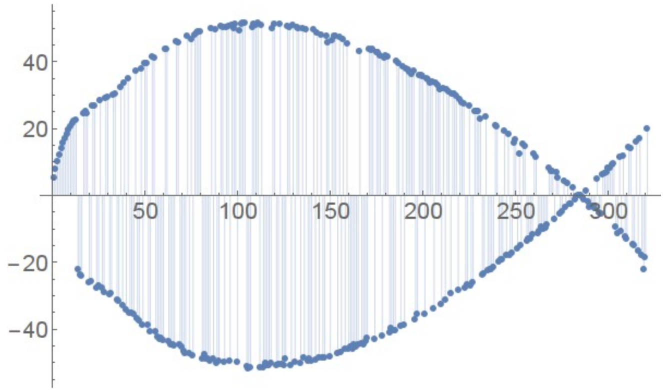
$\longrightarrow$ Problem: slow convergence.

Difficulties in the Evaluation



- Accuracy problems with standard floating-point arithmetic.

Difficulties in the Evaluation



- Accuracy problems with standard floating-point arithmetic.
- Use arbitrary-precision in a computer algebra system.
  But this makes computations even slower.

Holonomic Gradient Method (HGM)

$\longrightarrow$ Methods for evaluating and optimizing certain expressions.
(Nakayama, Nishiyama, Noro, Ohara, Sei, Takayama, Takemura)

Input: $f(x_1, \ldots, x_s)$ holonomic, $(a_1, \ldots, a_s) \in \mathbb{R}^s$
Output: an approximation of $f(a_1, \ldots, a_s)$

1. Determine a holonomic system (set of differential equations) to which f is a solution, and let r be its holonomic rank.
2. Determine a suitable "basis" of derivatives $\mathbf{f} = \left(f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}\right)$ of $f(x_1, \ldots, x_s)$.
3. Convert the holonomic system into a set of Pfaffian systems, i.e., $\frac{\mathrm{d}}{\mathrm{d}x_i}\mathbf{f} = \mathbf{A}_i\mathbf{f}$ for each x_i.
4. Compute $f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}$ at a suitably chosen point $(b_1, \ldots, b_s) \in \mathbb{R}^s$, for which this is easy to achieve.
5. Use your favourite numerical integration procedure (e.g., Euler, Runge-Kutta) to obtain $\mathbf{f}(a_1, \ldots, a_s)$.

Holonomic Functions

Definition. A function $f(x_1, \ldots, x_s)$ is **holonomic** if for each x_i it satisfies a linear differential equation with polynomial coefficients.

Holonomic Functions

Definition. A function $f(x_1, \ldots, x_s)$ is **holonomic** if for each x_i it satisfies a linear differential equation with polynomial coefficients.

Definition. A sequence $f(n_1, \ldots, n_s)$ is **holonomic** if its generating function

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} f(n_1, \ldots, n_s)\, x_1^{n_1} \cdots x_s^{n_s}$$

is holonomic in the above sense.

$\longrightarrow$ Definition applies also to the mixed case.

Holonomic Functions

Definition. A function $f(x_1, \ldots, x_s)$ is **holonomic** if for each x_i it satisfies a linear differential equation with polynomial coefficients.

Definition. A sequence $f(n_1, \ldots, n_s)$ is **holonomic** if its generating function

$$\sum_{n_1=0}^{\infty} \cdots \sum_{n_s=0}^{\infty} f(n_1, \ldots, n_s)\, x_1^{n_1} \cdots x_s^{n_s}$$

is holonomic in the above sense.

$\longrightarrow$ Definition applies also to the mixed case.

Fact. Holonomic functions are closed under addition, multiplication, (certain) substitutions, taking sums and integrals.

Example

The function $f(x) = {}_1F_1(a; b; x)$ is holonomic in x since it satisfies the differential equation

$$xf''(x) + (b - x)f'(x) - af(x) = 0.$$

In operator notation:

$$P(f) = 0 \quad \text{with} \quad P = xD_x^2 + (b - x)D_x - a.$$

Example

The function $f(x) = {}_1F_1(a; b; x)$ is holonomic in x since it satisfies the differential equation

$$xf''(x) + (b - x)f'(x) - af(x) = 0.$$

In operator notation:

$$P(f) = 0 \quad \text{with} \quad P = xD_x^2 + (b - x)D_x - a.$$

Likewise, ${}_1F_1(a; b; x)$ is holonomic w.r.t. the discrete variables a and b since

$$G(y, z) = \sum_{a=0}^{\infty} \sum_{b=0}^{\infty} {}_1F_1(a; b; x) y^a z^b$$

is holonomic in y and z.

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$ {}_1F_1(a; b; x) $$

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$ {}_1F_1\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right) $$

Substitution $a \to N, b \to n_2 + N_{\mathrm{R}}, x \to \dfrac{\Gamma_1 s}{1 - \Gamma_1 s x_1}$

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$ {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right) $$

$\dfrac{x_2^{n_2}}{n_2!}$ is holonomic (the generating function is $e^{x_2 y}$).

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\frac{x_2^{n_2}}{n_2!}\,{}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

Multiplication

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

Summation

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!}\, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

e^{-x_2} is holonomic.

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$e^{-x_2} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

Multiplication

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$e^{-x_2}\sum_{n_2=0}^{\infty}\frac{x_2^{n_2}}{n_2!}\,{}_1F_1\left(N; n_2+N_{\mathrm{R}}; \frac{\Gamma_1 s}{1-\Gamma_1 s x_1}\right)$$

$(1-\Gamma_1 s)^N$ is holonomic.

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\frac{e^{-x_2}}{(1-\Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

Division

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\frac{e^{-x_2}}{(1-\Gamma_1 s)^N}\sum_{n_2=0}^{\infty}\frac{x_2^{n_2}}{n_2!}\,{}_1F_1\!\left(N; n_2+N_{\mathrm{R}}; \frac{\Gamma_1 s}{1-\Gamma_1 s x_1}\right)$$

~~Division~~

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$\frac{e^{-x_2}}{(1-\Gamma_1 s)^N} \sum_{n_2=0}^{\infty} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s}{1 - \Gamma_1 s x_1}\right)$$

$(1-\Gamma_1 s)^{-N}$ is holonomic as well!

Closure Properties (Example)

We have seen that the following expression is holonomic:

$$M(s;x_1,x_2) = \frac{e^{-x_2}}{(1-\Gamma_1 s)^N}\sum_{n_2=0}^{\infty}\frac{x_2^{n_2}}{n_2!}\,{}_1F_1\!\left(N;n_2+N_{\mathrm{R}};\frac{\Gamma_1 s}{1-\Gamma_1 s x_1}\right)$$

Hence, by inspection, our SNR moment generating function is holonomic. Likewise, $p(t;x_1,x_2)$ and $P_{\mathrm{o}}(x_1,x_2)$ are holonomic.

Annihilator and Gröbner Bases

Definition. The set of all operators that annihilate a certain function is called its **annihilator**. It is a subset (more precisely: a **left ideal**) of the corresponding operator algebra.

Annihilator and Gröbner Bases

Definition. The set of all operators that annihilate a certain function is called its **annihilator**. It is a subset (more precisely: a **left ideal**) of the corresponding operator algebra.

Example. The annihilator of ${}_1F_1(a;b;x)$ in $\mathbb{C}(a,b,x)[S_a,S_b,D_x]$ is generated by the three operators

$$\begin{aligned}&(b-a)S_b + bD_x - b,\\ &aS_a - xD_x - a,\\ &xD_x^2 + (b-x)D_x - a.\end{aligned}$$

Annihilator and Gröbner Bases

Definition. The set of all operators that annihilate a certain function is called its **annihilator**. It is a subset (more precisely: a **left ideal**) of the corresponding operator algebra.

Example. The annihilator of ${}_1F_1(a;b;x)$ in $\mathbb{C}(a,b,x)[S_a,S_b,D_x]$ is generated by the three operators

$$
\begin{aligned}
&(b-a)S_b + bD_x - b,\\
&aS_a - xD_x - a,\\
&xD_x^2 + (b-x)D_x - a.
\end{aligned}
$$

Note: We use (left) Gröbner bases to deal with annihilators.

Pfaffian Systems

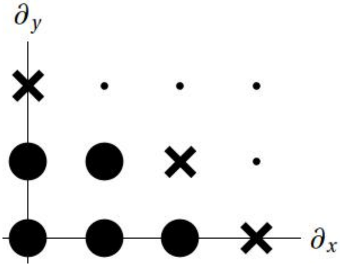
Fix $f(x_1, \ldots, x_s)$.

A suitable "basis of derivatives" $\mathbf{f} = \left(f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}\right)$ for HGM step 2 is given by the (finite!) list of monomials that are irreducible modulo the annihilator ideal.

Pfaffian Systems

Fix $f(x_1, \ldots, x_s)$.

A suitable "basis of derivatives" $\mathbf{f} = \left(f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}\right)$ for HGM step 2 is given by the (finite!) list of monomials that are irreducible modulo the annihilator ideal.



"monomials under the staircase" $(r = 5)$

Pfaffian Systems

Fix $f(x_1, \ldots, x_s)$.

A suitable "basis of derivatives" $\mathbf{f} = \left(f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}\right)$ for HGM step 2 is given by the (finite!) list of monomials that are irreducible modulo the annihilator ideal.

The **Pfaffian system** (given by the matrix $\mathbf{A}_i$) for x_i

$$\frac{\mathrm{d}}{\mathrm{d}x_i}\mathbf{f} = \mathbf{A}_i\mathbf{f}$$

is obtained by reduction with the Gröbner basis.

Pfaffian Systems

Fix $f(x_1, \ldots, x_s)$.

A suitable "basis of derivatives" $\mathbf{f} = \left(f^{(\mathbf{m}_1)}, \ldots, f^{(\mathbf{m}_r)}\right)$ for HGM step 2 is given by the (finite!) list of monomials that are irreducible modulo the annihilator ideal.

The **Pfaffian system** (given by the matrix $\mathbf{A}_i$) for x_i

$$\frac{\mathrm{d}}{\mathrm{d}x_i}\mathbf{f} = \mathbf{A}_i \mathbf{f}$$

is obtained by reduction with the Gröbner basis.

Note: For $s = 1$ (ODE case) the matrix $\mathbf{A}$ is a companion matrix.

Annihilator for $M(s; x_1, x_2)$

Apply creative telescoping (HolonomicFunctions package) to

$$\sum_{n_2=0}^{\infty} \frac{e^{-x_2}}{(1-\Gamma_1 s)^N} \frac{x_2^{n_2}}{n_2!} \, {}_1F_1\!\left(N; n_2 + N_{\mathrm{R}}; \frac{\Gamma_1 s x_1}{1-\Gamma_1 s}\right)$$

```
annM =
 CreativeTelescoping[Exp[-x2] / (1 - G1 * s) ^ N * x2 ^ n2 / n2 ! *
    Hypergeometric1F1[N, n2 + NR, G1 * s * x1 / (1 - G1 * s)],
   S[n2] - 1, {Der[s], Der[x1], Der[x2]}][[1]]
```

$\{(-s + G1\, s^2)\, D_s + x1\, D_{x1} + G1\, N\, s,$
$(-G1\, s\, x1\, x2 + x2^2 - G1\, s\, x2^2)\, D_{x2}^2 + (-NR\, x1 + G1\, NR\, s\, x1)\, D_{x1} +$
$(G1\, N\, s\, x1 - G1\, NR\, s\, x1 + NR\, x2 - G1\, NR\, s\, x2 - G1\, s\, x1\, x2 + x2^2 - G1\, s\, x2^2)$
$D_{x2} + G1\, N\, s\, x1,\ (G1\, s\, x1 - x2 + G1\, s\, x2)\, D_{x1}\, D_{x2} +$
$(-NR + G1\, NR\, s + G1\, s\, x1 - x2 + G1\, s\, x2)\, D_{x1} + G1\, N\, s\, D_{x2} + G1\, N\, s,$
$(G1\, s\, x1^2 - G1^2\, s^2\, x1^2 - x1\, x2 + 2\, G1\, s\, x1\, x2 - G1^2\, s^2\, x1\, x2)\, D_{x1}^2 +$
$(G1\, NR\, s\, x1 - G1^2\, NR\, s^2\, x1 - G1^2\, s^2\, x1^2 + G1\, s\, x1\, x2 - G1^2\, s^2\, x1\, x2)\, D_{x1} +$
$(-G1\, N\, s\, x2 + G1^2\, N\, s^2\, x2)\, D_{x2} - G1^2\, N\, s^2\, x1\}$

Annihilator for $p(t; x_1, x_2)$

```
ops = {Der[s], Der[t], Der[x1], Der[x2]};
annM1 = ToOrePolynomial[Prepend[annM, Der[t]], OreAlgebra @@ ops];
annp = CreativeTelescoping[
   DFiniteTimes[annM1, Annihilator[Exp[-s * t], ops]], Der[s]][[1]]
```

```
{ (G1 x1^2 x2 + 2 G1 x1 x2^2 + G1 x2^3) D_x2^2 +
   G1 NR t x1 D_t + (-G1 NR x1^2 - G1 NR x1 x2) D_x1 +
   (-G1 N x1^2 + G1 NR x1^2 - G1 N x1 x2 + 2 G1 NR x1 x2 +
     t x1 x2 + G1 x1^2 x2 + G1 NR x2^2 + 2 G1 x1 x2^2 + G1 x2^3) D_x2 +
   (G1 NR x1 - G1 N NR x1 + NR t x1 - G1 N x1^2 - G1 N x1 x2 + t x1 x2),
  (-G1 x1^2 - 2 G1 x1 x2 - G1 x2^2) D_x1 D_x2 + G1 NR t D_t +
   (-G1 NR x1 - G1 x1^2 - G1 NR x2 - 2 G1 x1 x2 - G1 x2^2) D_x1 +
   (-G1 N x1 - G1 N x2 + t x2) D_x2 +
   (G1 NR - G1 N NR + NR t - G1 N x1 - G1 N x2 + t x2),
  (G1 x1^3 + 2 G1 x1^2 x2 + G1 x1 x2^2) D_x1^2 +
   (G1 t x1^2 + G1 NR t x2 + 2 G1 t x1 x2 + G1 t x2^2) D_t +
   (G1 NR x1^2 + G1 x1^3 + G1 NR x1 x2 + 2 G1 x1^2 x2 + G1 x1 x2^2) D_x1 +
   (-G1 N x1 x2 - G1 N x2^2 + t x2^2) D_x2 + (G1 x1^2 + G1 NR x2 - G1 N NR x2 +
     ⋮
```

Annihilator for $P_\mathrm{o}(x_1, x_2)$

Recall:

$$P_\mathrm{o}(x_1, x_2) = \int_0^\tau p(t; x_1, x_2)\,\mathrm{d}t$$

Hence we apply creative telescoping to $p(t; x_1, x_2)$:

```
ct = CreativeTelescoping[annp, Der[t]]
```

$$\left\{\{D_{x2}, D_{x1}\},\ \left\{\frac{G1\,N\,t - t^2}{N\,x1} D_t + \frac{t}{N} D_{x1} - \frac{t}{N} D_{x2} + \frac{G1^2\,N - G1^2\,N^2 - G1\,t + 2\,G1\,N\,t - t^2}{G1\,N\,x1},\ \frac{G1\,t}{x1} D_t + \frac{G1 - G1\,N + t}{x1}\right\}\right\}$$

Annihilator for $P_o(x_1, x_2)$

```
OreGroebnerBasis[
 Flatten[
  MapThread[Function[{p, q},
    (# ** p) & /@ DFiniteSubstitute[DFiniteOreAction[annp, q],
      {t → τ}, Algebra → OreAlgebra[Der[x1], Der[x2]]]], ct]]]
```

$$\{-x1\, D_{x1} D_{x2} - x2\, D_{x2}^2 - x1\, D_{x1} + (-NR - x2)\, D_{x2},$$
$$(G1\, x1^2\, x2 + 2\, G1\, x1\, x2^2 + G1\, x2^3)\, D_{x2}^3 + G1\, NR\, x1^2\, D_{x1}^2 +$$
$$(G1\, x1^2 - G1\, N\, x1^2 + G1\, NR\, x1^2 + 3\, G1\, x1\, x2 - G1\, N\, x1\, x2 + 4\, G1\, NR\, x1\, x2 +$$
$$G1\, x1^2\, x2 + 2\, G1\, x2^2 + 2\, G1\, NR\, x2^2 + 2\, G1\, x1\, x2^2 + G1\, x2^3 + x1\, x2\, \tau)\, D_{x2}^2 +$$
$$(2\, G1\, NR\, x1^2 + G1\, NR\, x1\, x2)\, D_{x1} + (G1\, NR\, x1 - G1\, N\, NR\, x1 + 2\, G1\, NR^2\, x1 +$$
$$G1\, x1^2 - G1\, N\, x1^2 + G1\, NR\, x2 + G1\, NR^2\, x2 + 3\, G1\, x1\, x2 - G1\, N\, x1\, x2 +$$
$$2\, G1\, NR\, x1\, x2 + 2\, G1\, x2^2 + G1\, NR\, x2^2 + NR\, x1\, \tau + x1\, x2\, \tau)\, D_{x2},$$
$$(-G1\, x1^4 - 2\, G1\, x1^3\, x2 - G1\, x1^2\, x2^2)\, D_{x1}^3 + (-G1\, x1^3 - G1\, NR\, x1^3 - 2\, G1\, x1^4 -$$
$$2\, G1\, x1^2\, x2 - 2\, G1\, NR\, x1^2\, x2 - 4\, G1\, x1^3\, x2 - G1\, x1\, x2^2 - 2\, G1\, x1^2\, x2^2)\, D_{x1}^2 +$$
$$(-G1\, x1^3\, x2 - G1\, x1\, x2^2 - G1\, N\, x1\, x2^2 - G1\, NR\, x1\, x2^2 -$$
$$2\, G1\, x1^2\, x2^2 - G1\, x2^3 - G1\, N\, x2^3 - G1\, x1\, x2^3 + x2^3\, \tau)\, D_{x2}^2 +$$
$$(-G1\, x1^3 - G1\, N\, x1^3 - G1\, x1^4 - 2\, G1\, x1^2\, x2 - 2\, G1\, N\, x1^2\, x2 -$$

⋮

HGM Computation

The irreducible monomials of the annihilator of $P_{\rm o}(x_1, x_2)$ are

$$1, D_1, D_2, D_1^2, D_2^2.$$

Hence, we take the following basis:

$$\mathbf{f} = \left(P_{\rm o}, P_{\rm o}^{(0,1)}, P_{\rm o}^{(1,0)}, P_{\rm o}^{(2,0)}, P_{\rm o}^{(0,2)}\right).$$

HGM Computation

The irreducible monomials of the annihilator of $P_{\rm o}(x_1, x_2)$ are

$$1, D_1, D_2, D_1^2, D_2^2.$$

Hence, we take the following basis:

$$\mathbf{f} = \left(P_{\rm o}, P_{\rm o}^{(0,1)}, P_{\rm o}^{(1,0)}, P_{\rm o}^{(2,0)}, P_{\rm o}^{(0,2)}\right).$$

Initial values are computed for some small x_1, x_2, so that the infinite series converges quickly.

HGM Computation

The irreducible monomials of the annihilator of $P_{\mathrm{o}}(x_1, x_2)$ are

$$1, D_1, D_2, D_1^2, D_2^2.$$

Hence, we take the following basis:

$$\mathbf{f} = \left(P_{\mathrm{o}}, P_{\mathrm{o}}^{(0,1)}, P_{\mathrm{o}}^{(1,0)}, P_{\mathrm{o}}^{(2,0)}, P_{\mathrm{o}}^{(0,2)}\right).$$

Initial values are computed for some small x_1, x_2, so that the infinite series converges quickly.

The matrix $\mathbf{A}_1$ of the Pfaffian system $D_1\mathbf{f} = \mathbf{A}_1\mathbf{f}$ is

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & -\frac{N_{\mathrm{R}}+x_2}{x_1} & -1 & -\frac{x_2}{x_1} & 0 \\ 0 & \langle\cdots\rangle & -\frac{N_{\mathrm{R}}x_1(2x_1+x_2)}{x_2(x_1+x_2)^2} & \langle\cdots\rangle & -\frac{N_{\mathrm{R}}x_1^2}{x_2(x_1+x_2)^2} \\ 0 & \langle\cdots\rangle & \frac{N_{\mathrm{R}}x_1}{(x_1+x_2)^2} & \langle\cdots\rangle & -\frac{(x_1+x_2)^2+N_{\mathrm{R}}x_2}{(x_1+x_2)^2} \end{pmatrix}.$$

HGM Computation

The irreducible monomials of the annihilator of $P_{\mathrm{o}}(x_1, x_2)$ are

$$1, D_1, D_2, D_1^2, D_2^2.$$

Hence, we take the following basis:

$$\mathbf{f} = \left(P_{\mathrm{o}}, P_{\mathrm{o}}^{(0,1)}, P_{\mathrm{o}}^{(1,0)}, P_{\mathrm{o}}^{(2,0)}, P_{\mathrm{o}}^{(0,2)}\right).$$

Initial values are computed for some small x_1, x_2, so that the infinite series converges quickly.

The matrix $\mathbf{A}_1$ of the Pfaffian system $D_1\mathbf{f} = \mathbf{A}_1\mathbf{f}$ is obtained by rewriting $P_{\mathrm{o}}^{(1,0)}, P_{\mathrm{o}}^{(1,1)}, P_{\mathrm{o}}^{(2,0)}, P_{\mathrm{o}}^{(3,0)}, P_{\mathrm{o}}^{(1,2)}$ in terms of $\mathbf{f}$.
Similar for $D_2\mathbf{f} = \mathbf{A}_2\mathbf{f}$.

HGM Computation

The irreducible monomials of the annihilator of $P_{\mathrm{o}}(x_1, x_2)$ are

$$1, D_1, D_2, D_1^2, D_2^2.$$

Hence, we take the following basis:

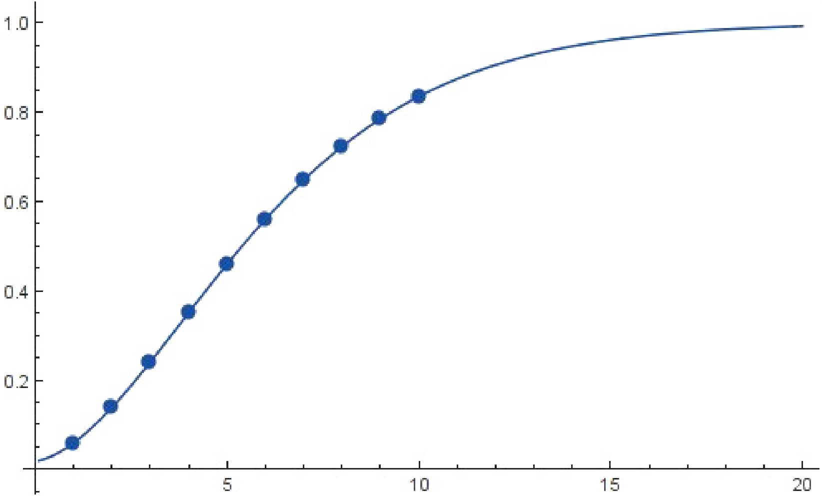
$$\mathbf{f} = \big(P_{\mathrm{o}}, P_{\mathrm{o}}^{(0,1)}, P_{\mathrm{o}}^{(1,0)}, P_{\mathrm{o}}^{(2,0)}, P_{\mathrm{o}}^{(0,2)}\big).$$

Initial values are computed for some small x_1, x_2, so that the infinite series converges quickly.

The matrix $\mathbf{A}_1$ of the Pfaffian system $D_1\mathbf{f} = \mathbf{A}_1\mathbf{f}$ is obtained by rewriting $P_{\mathrm{o}}^{(1,0)}, P_{\mathrm{o}}^{(1,1)}, P_{\mathrm{o}}^{(2,0)}, P_{\mathrm{o}}^{(3,0)}, P_{\mathrm{o}}^{(1,2)}$ in terms of $\mathbf{f}$.
Similar for $D_2\mathbf{f} = \mathbf{A}_2\mathbf{f}$.

$\mathbf{A}_1$ and $\mathbf{A}_2$ allow to propagate the initial values along both coordinate axes.

HGM Computation



- dots: computed with truncated series $(167s)$
- line: computed with HGM $(< 1s)$

Application

"From an engineering perspective, Fig. 6 reveals that even at Γ_b as low as 0 dB (i.e., low transmit power, or strong receiver noise), the low-complexity ZF can sustain a rate of almost 5 bits per channel use (bpcu) for each of the 20 streams, if the base station has 100 antennas. If a WiFi system would implement this, then, within only 20 MHz of bandwidth it would support 20 users each of them downloading at 20 MHz $\times$ 5 bpcu $=$ 100 Mbps. But nowadays WiFi routers already can use 100 MHz channels (in the 5GHz band), which would bring the download speed for each user at 0.5 Gbps. This would be very useful in a conference hall ;-) "

Outlook

- consider more complicated models
- choice of direction for numerical integration
- certified numerical evaluations