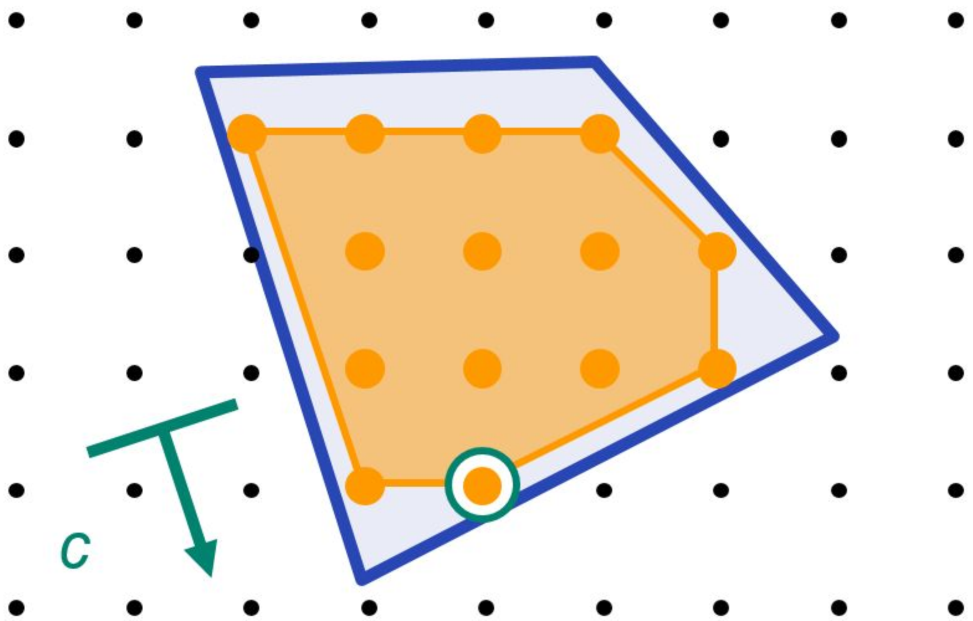
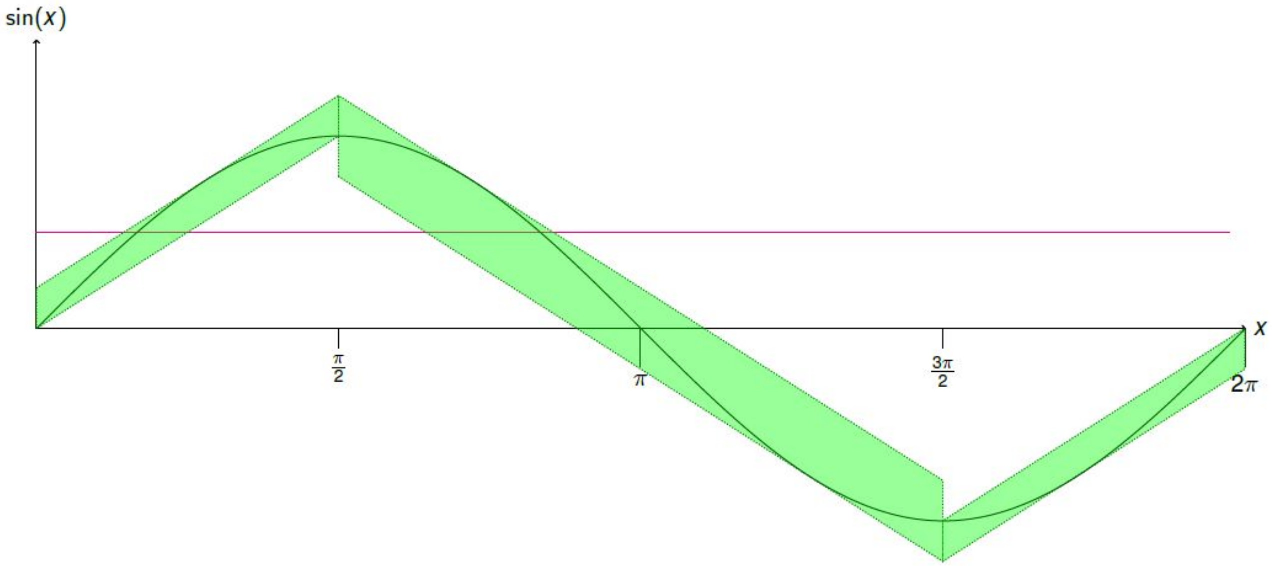
MIXED INTEGER OPTIMIZATION PROBLEMS ON NETWORKS WITH PDE CONSTRAINTS

Trends in Mathematical Sciences Conference 2024, June 10th 2024, Erlangen

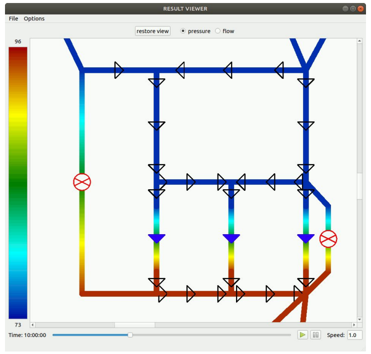
Alexander Martin, Technische Universität Nürnberg (UTN) & Fraunhofer IIS

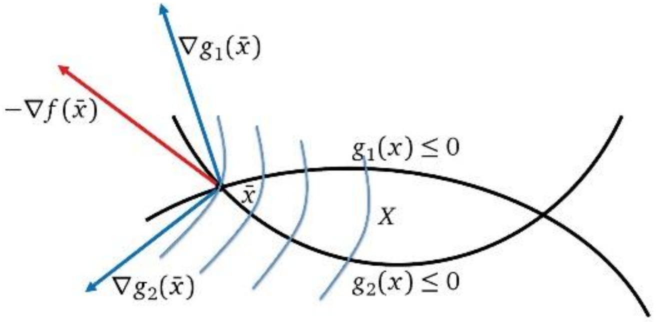


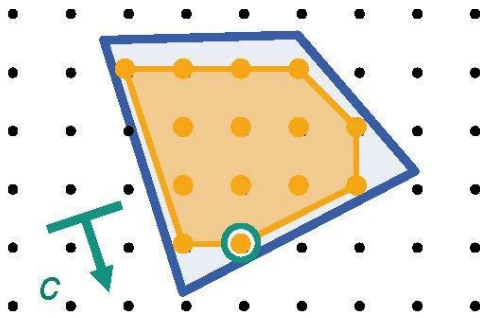


TRR 154 UTN FAU Friedrich-Alexander-Universität Erlangen-Nürnberg Fraunhofer IIS

Mathematical modeling, simulation, and optimization







Modeling and numerical simulation

- existence, uniqueness, regularity
- efficient algorithms, convergence, error control

Nonlinear optimization and control

- efficient algorithms, convergence, error control
- local optima and their characterization

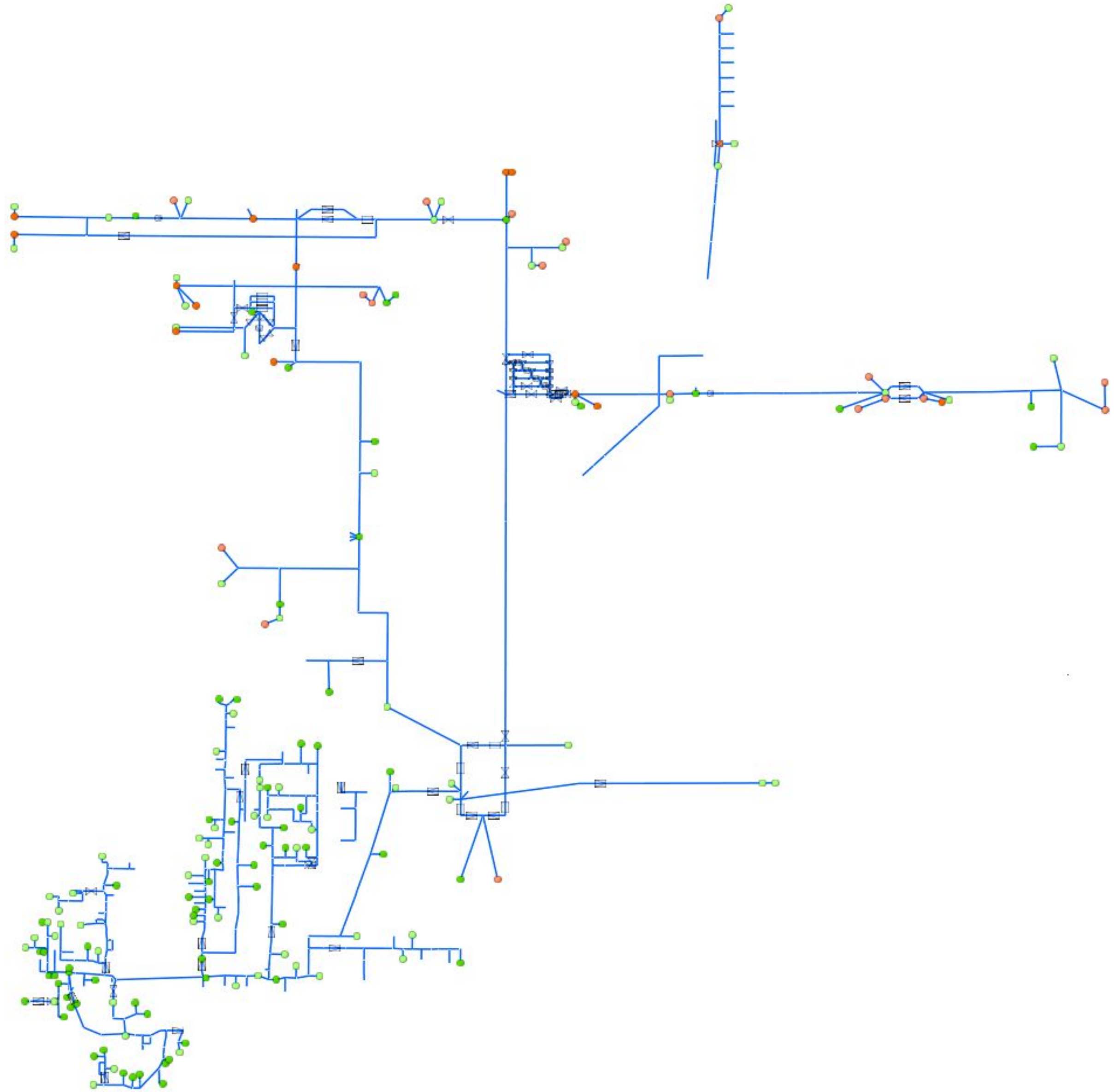
Challenge: Coupling integer and continuous

- active research area
- general methods out-of-reach

A typical network flow problem

- 661 vertices
- 689 arcs
- 32 sources
- 142 sinks

except that the "flow" is gas



Gas networks

- Physics are inherently continuous

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho v)}{\partial x} = 0$$

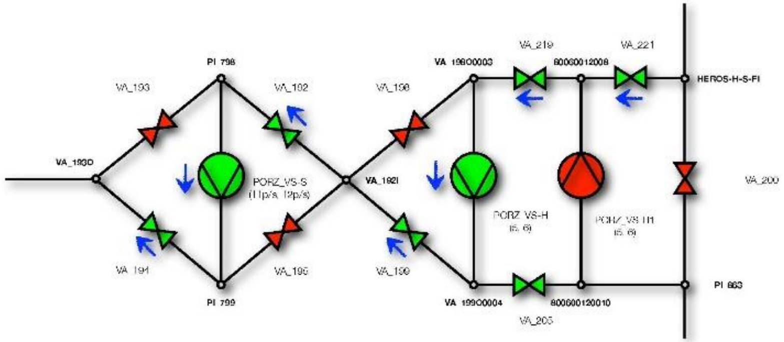
$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho v^2 + p)}{\partial x} + g\rho\frac{\partial h}{\partial x} + \frac{\lambda}{2D}\rho\,|v|\,v = 0$$

$$\frac{\partial E}{\partial t} + \frac{\partial(Ev + pv)}{\partial x} + A\rho v g\frac{\partial h}{\partial x} + \pi\, D\, c_{\mathrm{HT}}\,(T - T_{\mathrm{soil}}) = 0.$$

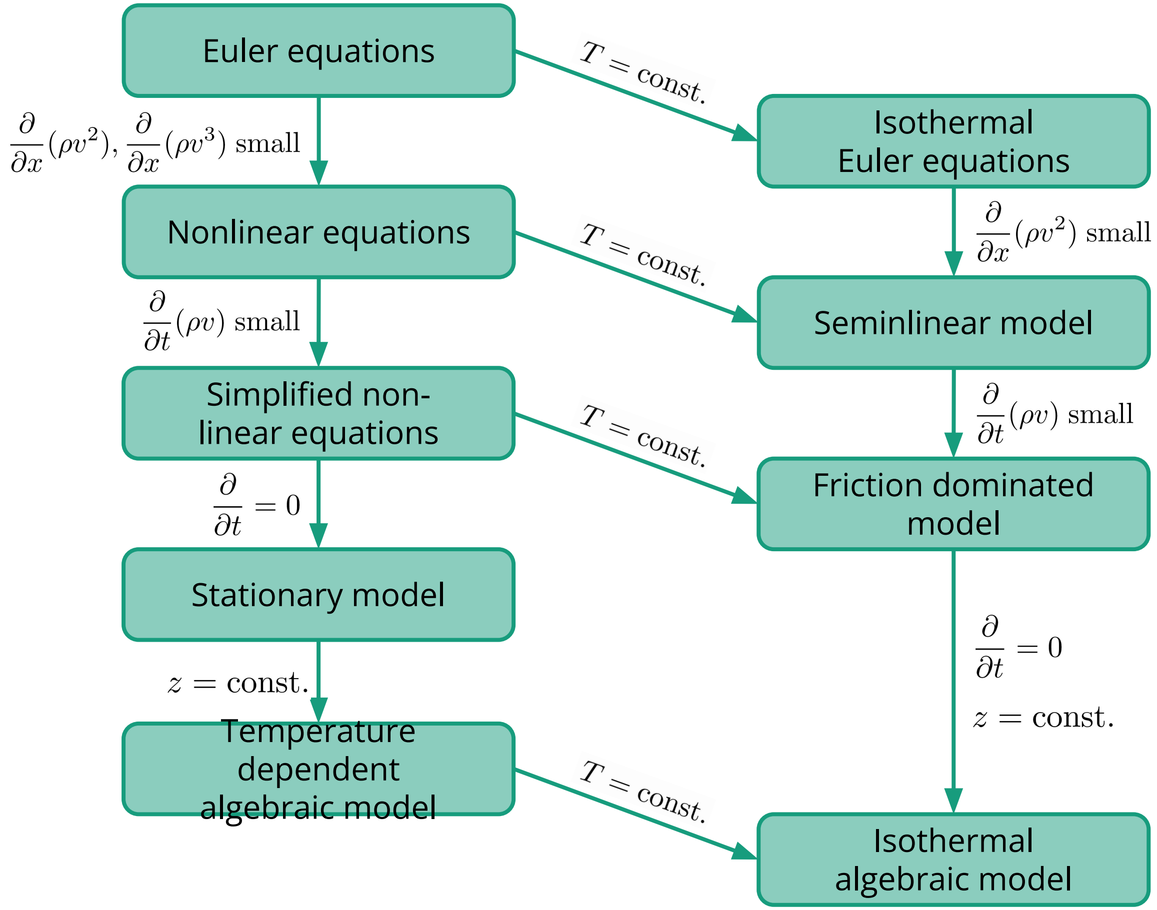
- Networks are inherently discrete with edges that can be switched on or off

Valve a with switching variable $s_a \in \{0, 1\}$

$$s_a = 0 \;\Rightarrow\; q_a = 0$$
$$s_a = 1 \;\Rightarrow\; p_i = p_j$$



A model hierarchy for pipes



MINLP model

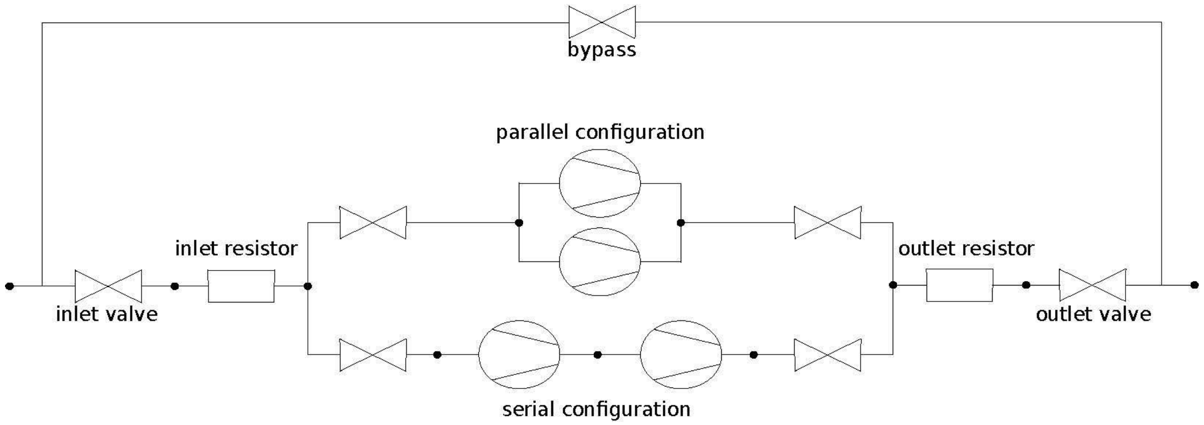
- Variables
  - Pressure p_i for nodes $i \in \mathbb{V}$
  - Standard volumetric flow q_a for arcs $a \in \mathbb{A}$
  - Compressor power P_a for compressors $a \in \mathbb{A}$
- Nonlinear constraints
  - Pressure loss over a pipe $a = (i, j)$: $p_j^2 = \left(p_i^2 - \Lambda_a |q_a| q_a \frac{e^{S_a}-1}{S_a}\right) e^{-S_a}$
  - Pressure loss over a resistor $a = (i, j)$:

    $$p_i^2 - p_j^2 + |\Delta_{ij}|\Delta_{ij} = \frac{16\rho_0 p_0 z_m}{\pi^2 z_0 T_0}\frac{\xi_a T}{D_a^4}|q_a|q_a, \quad \Delta_{ij} = p_i - p_j$$
  - Power consumption of a compressor unit $a = (i, j)$:

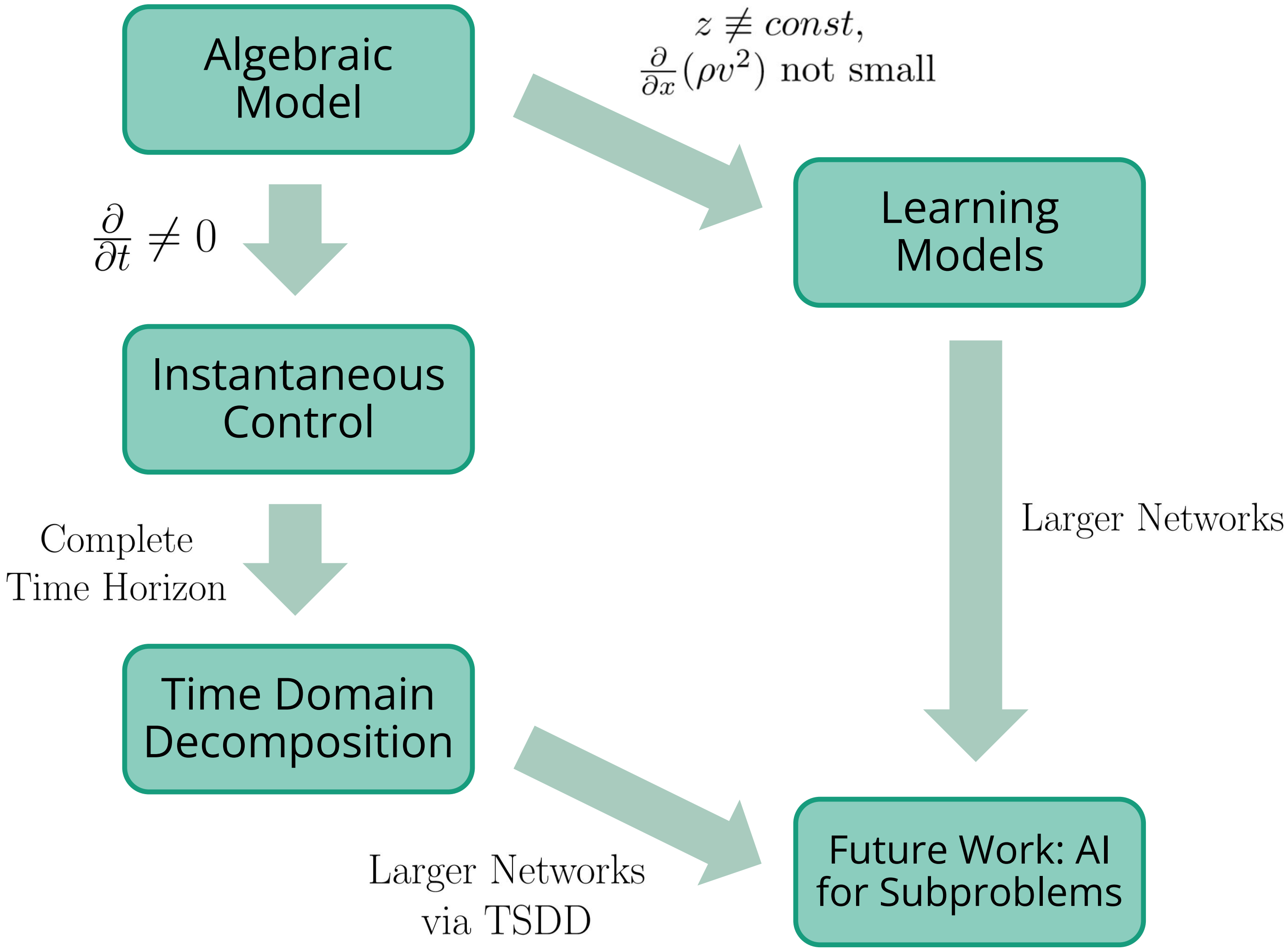
    $$P_a = \frac{\kappa}{\kappa-1}\frac{\rho_0 R T_i z_i}{\eta_{ad,a} m}\left(\left(\frac{p_j}{p_i}\right)^{\frac{\kappa-1}{\kappa}} - 1\right) q_a$$

MINLP model

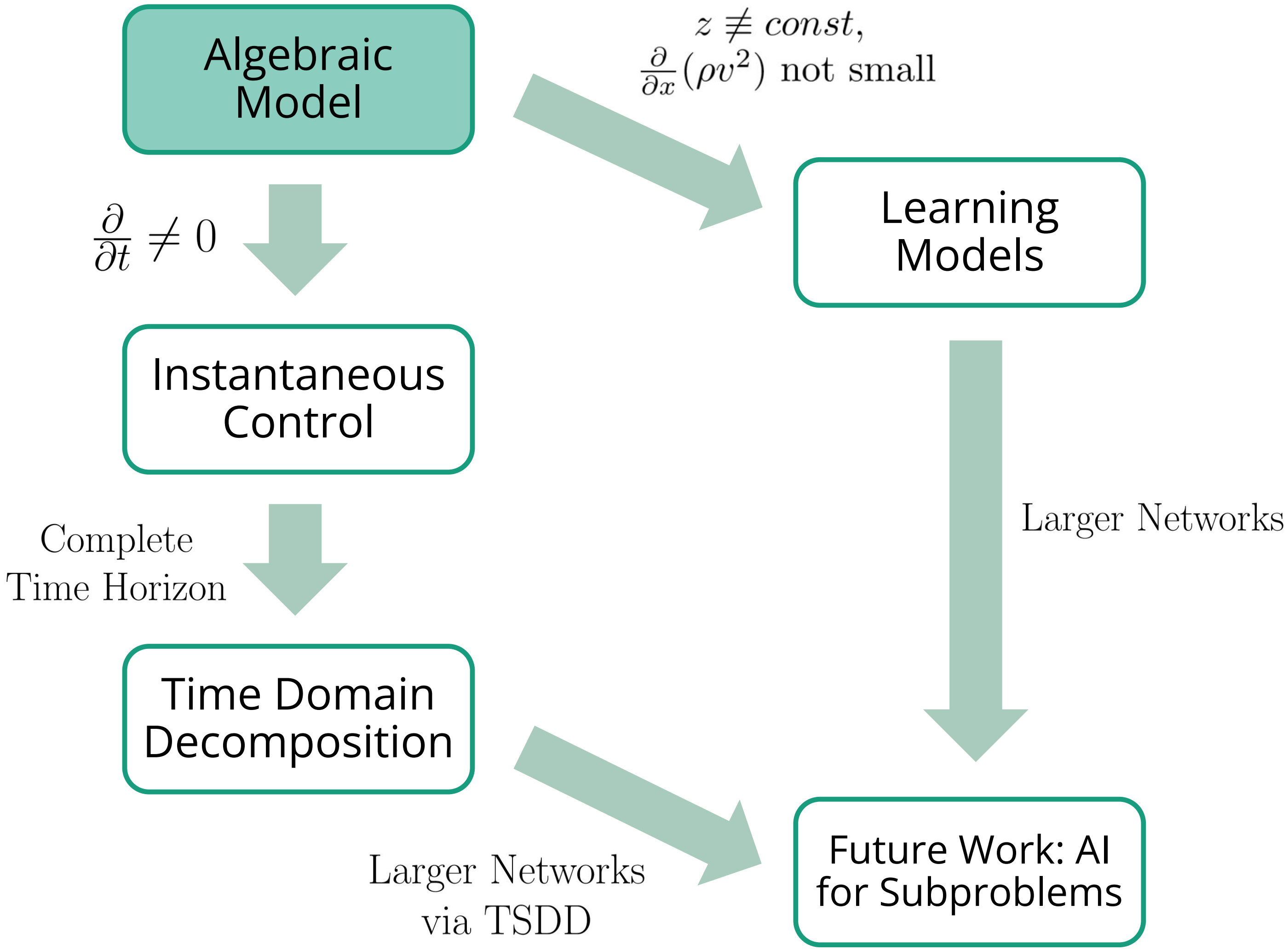
- Switching variables $s_a \in \{0,1\}$ for active elements $a \in \mathbb{A}$
- Combinatorial constraints:
  - Valve: $s_a = 0 \rightarrow q_a = 0$, $s_a = 1 \rightarrow p_i = p_j$
  - Control valves: : $s_a = 0 \rightarrow q_a = 0$, $s_a = 1 \rightarrow p_i = p_j \in [\Delta_a^-, \Delta_a^+]$
  - Compressors: $s_a = 0 \rightarrow q_a = 0$, $s_a = 1 \rightarrow p_j = f(P_a, q_a, p_i)$
  - Configurations: $\sum_{c \in C} s_c + s_{bypass} \leq 1$



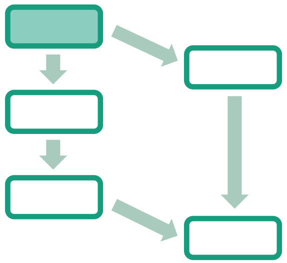
Overview


Algebraic Model
$z \not\equiv const,$
$\frac{\partial}{\partial x}(\rho v^2)$ not small
Learning Models
$\frac{\partial}{\partial t} \neq 0$
Instantaneous Control
Larger Networks
Complete Time Horizon
Time Domain Decomposition
Larger Networks via TSDD
Future Work: AI for Subproblems

Overview


Algebraic Model
$z \neq const,$
$\frac{\partial}{\partial x}(\rho v^2)$ not small
Learning Models
$\frac{\partial}{\partial t} \neq 0$
Instantaneous Control
Complete Time Horizon
Larger Networks
Time Domain Decomposition
Larger Networks via TSDD
Future Work: AI for Subproblems

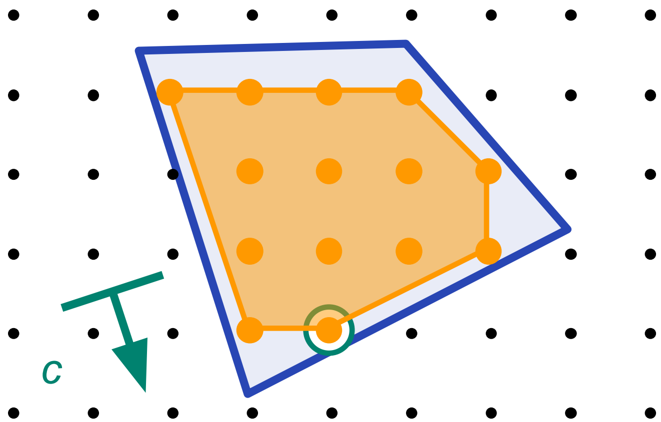
Algebraic model - A Mixed Integer Program (MIP)



Algebraic Model

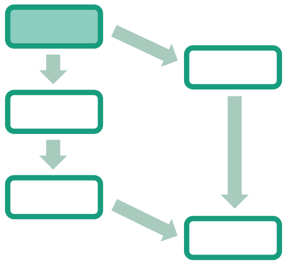
$$\min \quad c^T x$$

$$\begin{aligned} &(i) \quad & Ax &\leq b \\ &(ii) \quad & x &\in \mathbb{Z}^p \times \mathbb{R}^{n-p} \end{aligned}$$

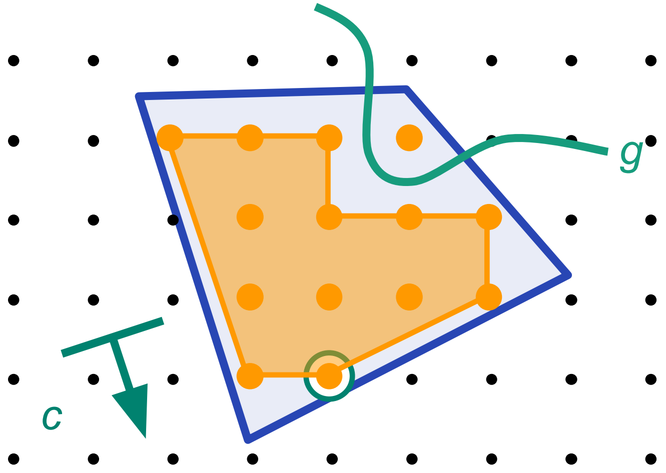


- p = 0: linear program
- p = n: integer program
- 0 < p < n: mixed integer program

Algebraic model - A Mixed Integer Non-Linear Program (MINLP)



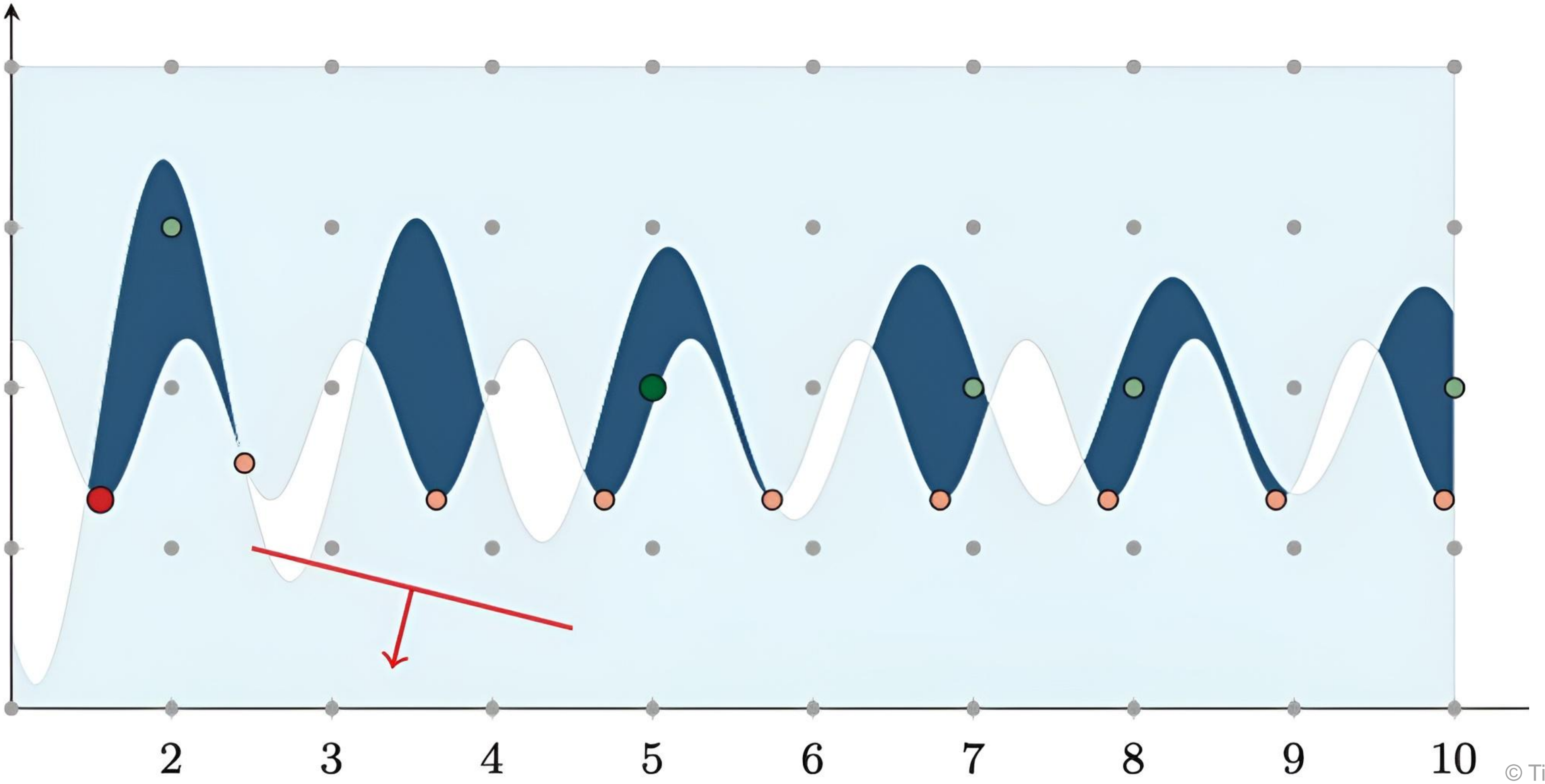
$$\begin{aligned} \min \quad & c^T x \\ \text{(i)} \quad g(x) & \leq 0 \\ \text{(ii)} \quad Ax & \leq b \\ \text{(iii)} \quad x & \in \mathbb{Z}^p \times \mathbb{R}^{n-p} \end{aligned}$$



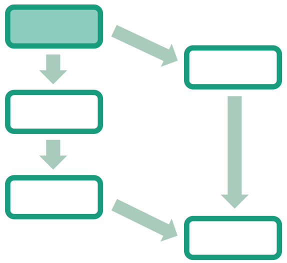
- p = 0: non-linear program
- p = n: integer non-linear program
- 0 < p < n: mixed integer nonlinear program

with some non-linear function $g: R^n \rightarrow R^m$

A "Simple" Mixed Integer Nonlinear Programming



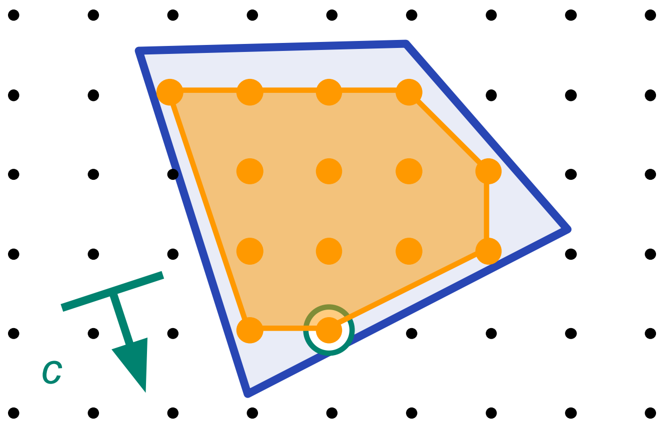
Algebraic model - A Mixed Integer Program (MIP)



Algebraic Model

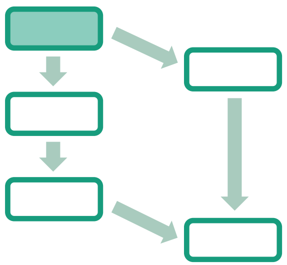
$$\min \quad c^T x$$

$$\begin{array}{lrcl} \text{(i)} & Ax & \leq & b \\ \text{(ii)} & x & \in & \mathbb{Z}^p \times \mathbb{R}^{n-p} \end{array}$$



- p = 0: linear program
- p = n: integer program
- 0 < p < n: mixed integer program

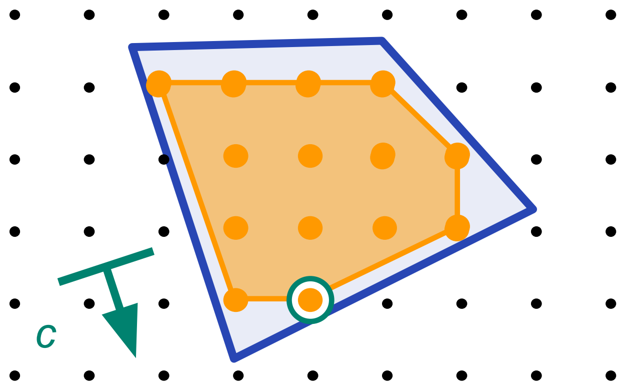
Algebraic model - Branch-and-Cut Method for MIPs at a Glance



Algebraic Model

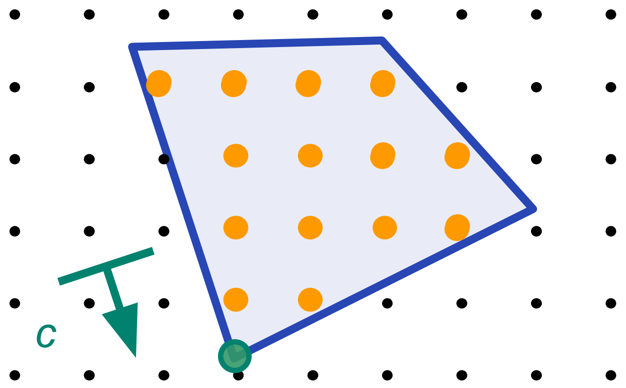
(1) Modelling the problem as a MIP

$$\begin{aligned} \min \quad & c^T x \\ \text{(i)} \quad Ax & \leq b \\ \text{(ii)} \quad x & \in \mathbb{Z}^p \times \mathbb{R}^{n-p} \end{aligned}$$



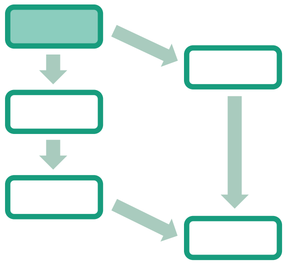
(2) Solve the LP-Relaxation

$$\begin{aligned} \min \quad & c^T x \\ \text{(i)} \quad Ax & \leq b \\ \text{(ii)} \quad x & \in \mathbb{R}^n \end{aligned}$$



Let x^* be an optimal solution.

Algebraic model - Branch-and-Cut Method for MIPs at a Glance

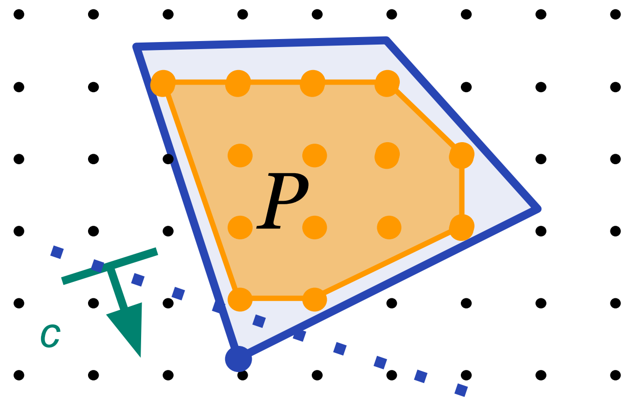


Algebraic Model

(3) Generate Cutting Planes

- Describe $\bar{P}$ by inequalities
- Find $a^T x \le \alpha$ valid for $\bar{P}$

  with $a^T x > \alpha$ (separation)

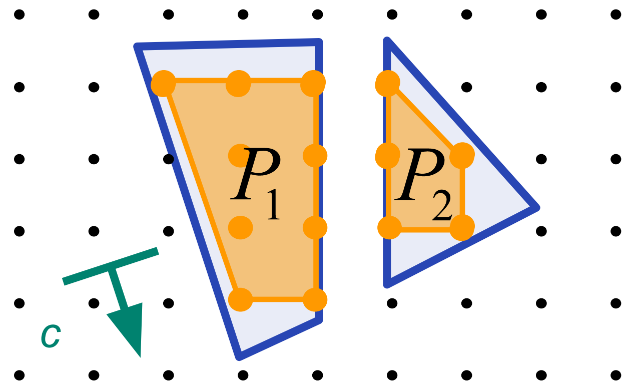


(4) Branch and Bound

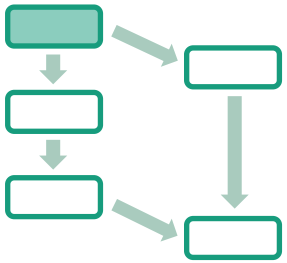
- Choose j with $x_j^* \notin \mathbb{Z}$
- Split the problem in

$$P_1 = P \cap \{x : x_j \geq \lceil x_j^* \rceil\}$$

and $P_2 = P \cap \{x : x_j \leq \lfloor x_j^* \rfloor\}$



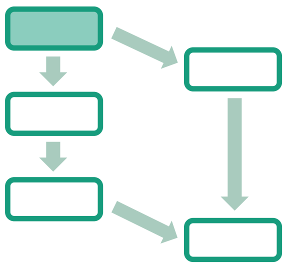
Algebraic model – Solution methods for MINLPs



Algebraic Model

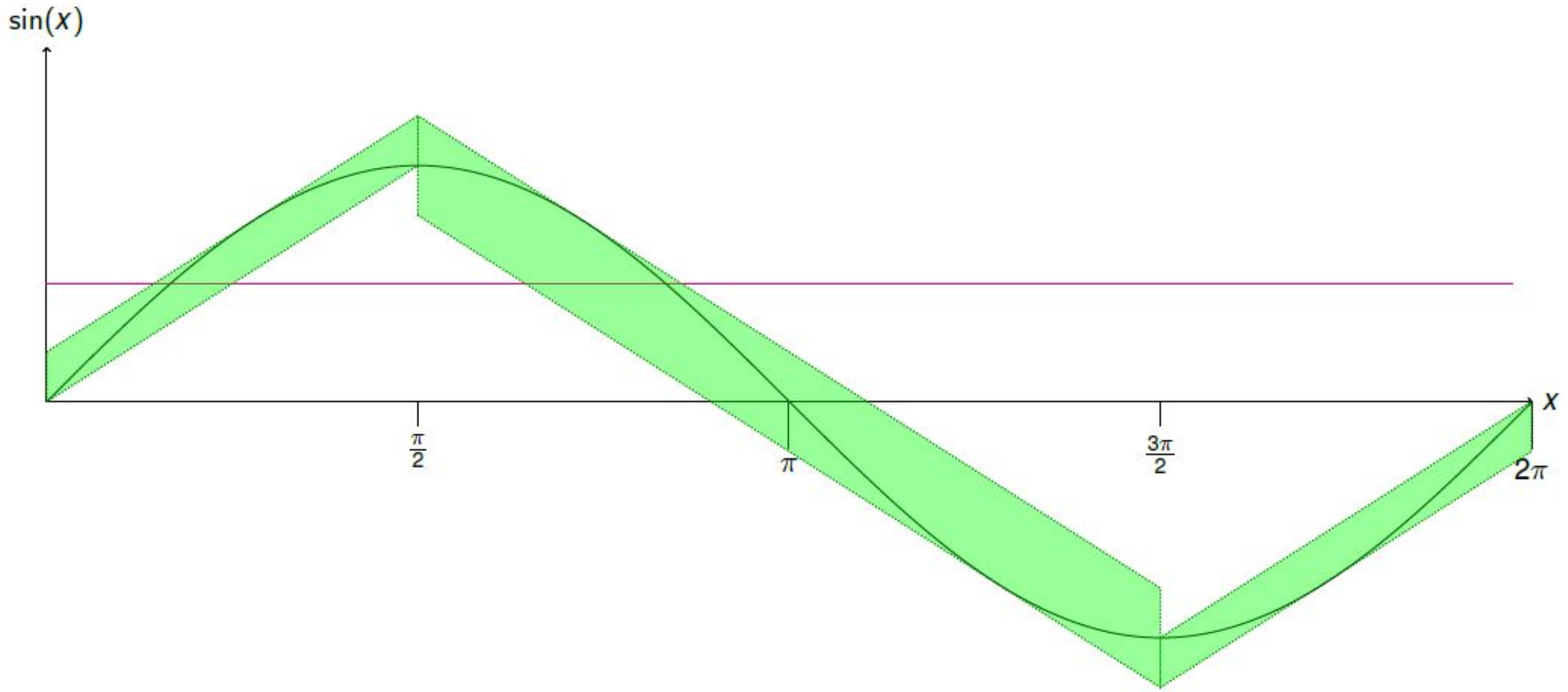
- Such problems are typicalley solved by outer approximation and spatial branching
  - Baron (Tawarmalani, Sahinidis 2005)
  - Couenne (Belotti, Lee, Liberti, Margot, Wächter 2009)
  - SCIP (Vigerske 2013)
  - alphaECP (Westerlund, Lindquist 2003)
  - Bonmin (Bonami, Biegler, Conn, Cornuejols, Grossmann, Laird, Lee, Lodi, Margot, Sawaya, Wächter 2005)
  - ...
- In case of nonconvex (MI)NLP spatial branching is unavoidable
- Lots of branching experiences for (M)IPs
- Polyhedral combinatorics help to avoid parts of branching

Algebraic model - Example

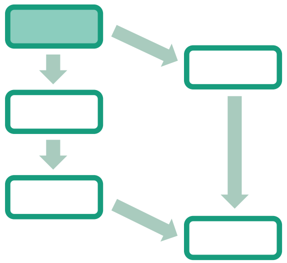


Algebraic Model

e. g. sin(*x*) = 0,5



Algebraic model – Consistent hierarchical modeling approach



Algebraic Model

(1) Determine relaxation error ε

(2) Set up the MIP relaxation with accuracy ε

(3) Solve the MIP relaxation

(4) If MIP is infeasible ! STOP
(MINLP is infeasible)

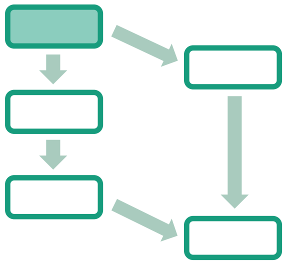
(5) Fix the discrete decision variables in the MINLP model according to the MIP solution

(6) Solve the remaining NLP model

(7) If NLP is feasible ! STOP
(feasible MINLP solution found; solution quality within ε)

(8) Reduce ε, goto (2)

Algebraic model – Consistent hierarchical modeling approach



Algebraic Model

(1) Determine relaxation error ε

(2) Set up the MIP relaxation with accuracy ε

(3) Solve the MIP relaxation

(4) If MIP is infeasible ! STOP
(MINLP is infeasible)

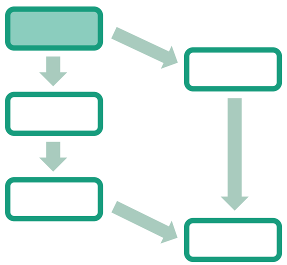
(5) Fix the discrete decision variables in the MINLP model according to the MIP solution

(6) Solve the remaining NLP model

(7) If NLP is feasible ! STOP
(feasible MINLP solution found; solution quality within ε)

(8) Reduce ε, goto (2)

Algebraic model - Adaptive piecewise linear interpolation



Algebraic Model

Algorithm 1: Adaptive piecewise linear interpolation

Data: A convex polytope $\mathcal{P} \subseteq \mathbb{R}^d$, a continuous function $f : \mathcal{P} \to \mathbb{R}$ and an upper bound $\epsilon > 0$ for the approximation error.

Result: A triangulation $\mathcal{S}$ of $\mathcal{P}$ corresponding to a piecewise lineare interpolation ϕ of f over $\mathcal{P}$ with $\phi(\mathbf{x}) = f(\mathbf{x})$ for all $\mathbf{x} \in \mathcal{V}(\mathcal{S})$ and $\phi = \phi_i$ for $S_i \in \mathcal{S}$ and $i = 1, \ldots, n$.

Set $\mathcal{V} = \mathcal{V}(\mathcal{P})$;

Construct an initial triangulation $\mathcal{S}$ of $\mathcal{V}$ and the corresponding piecewise linear interpolation ϕ of f with $\phi(\mathbf{x}) = \phi_i(\mathbf{x})$ for $\mathbf{x} \in S_i$ for all $S_i \in \mathcal{S}$;

while $\exists S_i \in \mathcal{S}$, S_i *unmarked* **do**

  if $\epsilon(f, S) := \max_{\mathbf{x} \in S_i} |f(\mathbf{x}) - \phi_i(\mathbf{x})| > \epsilon$ **then**

    Add a point, where the maximal error is attained to $\mathcal{V}$;

    Set $\mathcal{S} \leftarrow \mathcal{S} \setminus \{S_i\}$;

    Update $\mathcal{S}$ according to the extended set of vertices $\mathcal{V}$;

  else

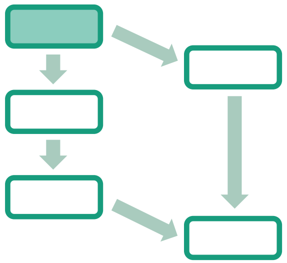
    Mark S_i;

  end

end

return $\mathcal{S}$

Algebraic model - Computing the approximation error



Algebraic Model

- If f is convex or concave over S, this is easy!
- f f is indefinite over S computing (f ; S) requires the solution of nonconvex NLPs to global optimality (in general NP-hard, cf. Murty & Kabadi 1985)

Definition

A function

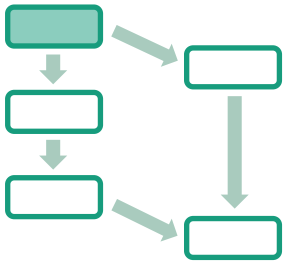
$$\mu \in \mathcal{U}(f, S) := \{\xi : S \to \mathbb{R} \ : \ \xi \text{ convex}, \ \xi(\mathbf{x}) \leq f(\mathbf{x}) \ \forall \mathbf{x} \in S\}$$

is called *convex underestimator* of f over S. The function $\mathrm{vex}_S[f] : S \to \mathbb{R}$ defined as

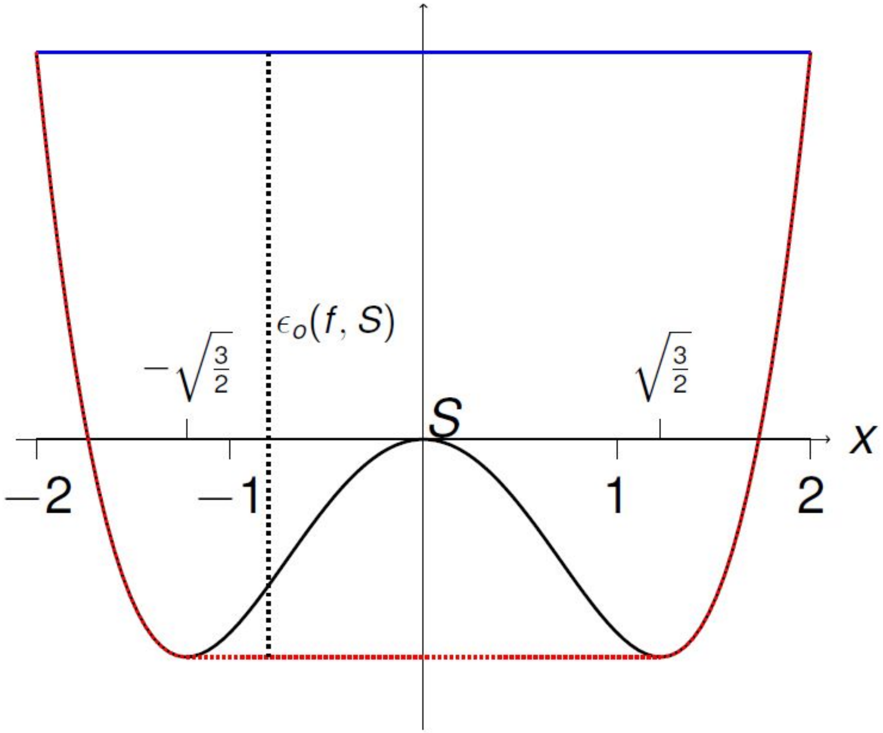
$$\mathrm{vex}_S[f](\mathbf{x}) := \sup\{\mu(\mathbf{x}) \ : \ \mu \in \mathcal{U}(f, S)\}$$

is called convex envelope of f over S.

Algebraic model - Computing the approximation error

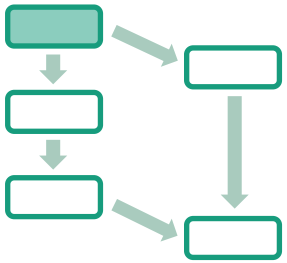


$f(x) = x^4 - 3x^2$, $\mathrm{vex}_S[f](x)$, $\phi(x)$



$$\mathcal{M}_0 = \left\{-\sqrt{\tfrac{3}{2}}, \sqrt{\tfrac{3}{2}}\right\}, \; \mathcal{N}_0 = \left[-\sqrt{\tfrac{3}{2}}, \sqrt{\tfrac{3}{2}}\right]$$

Algebraic model - Computing the approximation error



Algebraic Model

Theorem (Gugat, Martin, Morsi, Schewe 2011)

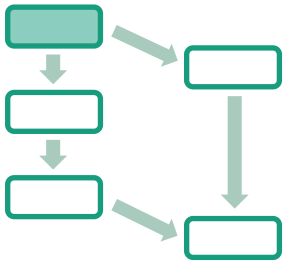
Let $\mathcal{M}_o$ be the set of global maximizers for the overestimation of f by ϕ over S and let $\mathcal{N}_o$ be the set of points, where the global maximum of the overestimation of the convex envelope of f over S by ϕ is attained. Then,

$$\epsilon(f, S) = \epsilon(\mathrm{vex}_S[f], S) \text{ and } \mathcal{N}_o = \mathrm{conv}(\mathcal{M}_o).$$

Theorem (Gugat, Martin, Morsi, Schewe 2011)

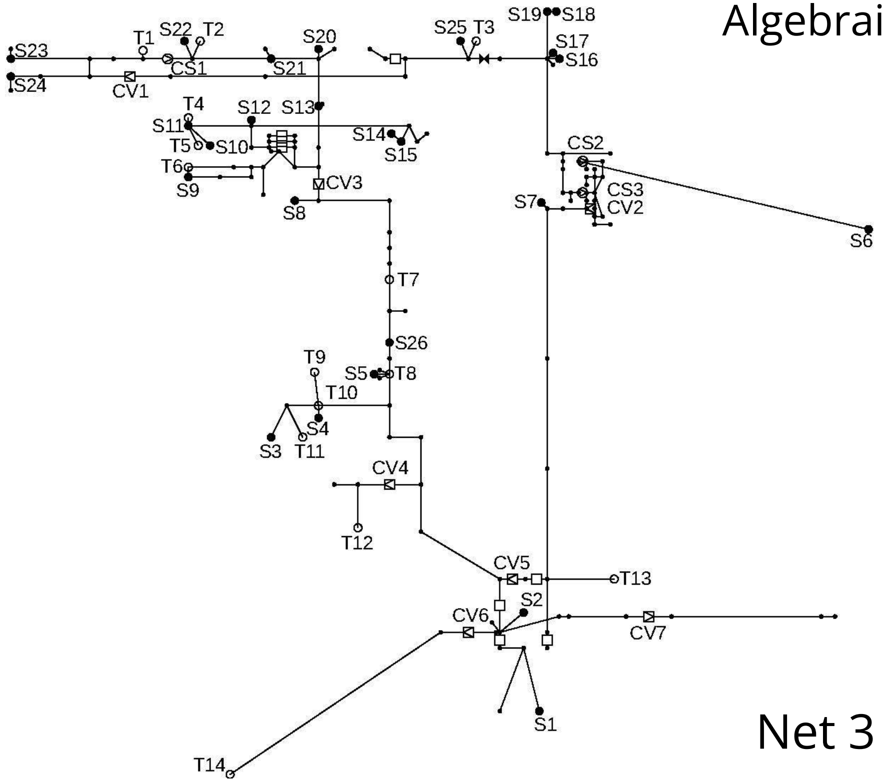
Let ϕ be the linear interpolation of f over a d-simplex S. Then a point $\mathbf{x}^* \in \mathcal{M}_o$ can be obtained by solving at most d convex optimization problems in dimension $\leq d$.

Numerical results – small (real) instances



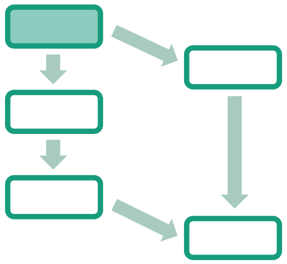
Algebraic Model

net	ϵ	cont	bin	cons	t_{MIP}	feas	t_{NLP}
1	10.0	377	42	685	0.01s	y	0.11s
1	5.0	380	45	694	0.01s	y	0.15s
1	2.5	387	52	714	0.01s	y	0.06s
1	1.0	423	88	823	0.02s	y	0.20s
2	10.0	450	67	859	0.05s	y	0.26s
2	5.0	479	96	946	0.09s	y	0.20s
2	2.5	543	160	1138	0.11s	y	0.09s
2	1.0	816	433	1957	0.13s	y	0.26s
3	10.0	2099	418	3868	1.24s	n	12.94s
3	5.0	2412	713	4807	1.51s	y	1.48s
3	2.5	3058	1377	6745	6.03s	y	1.32s
3	1.0	5185	3504	13126	22.04s	y	1.59s
4	10.0	4825	1663	10994	21.65s	n	41.33s
4	5.0	6012	2850	14555	51.26s	y	30.83s
4	2.5	8433	5217	21818	132.96s	y	36.65s
4	1.0	16343	13181	45548	600.00s	-	-



Net 3

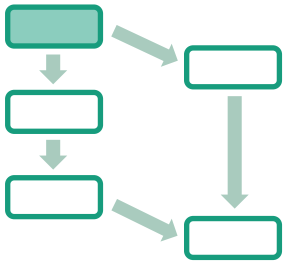
Numerical results – small (real) instances: a comparison



Algebraic Model

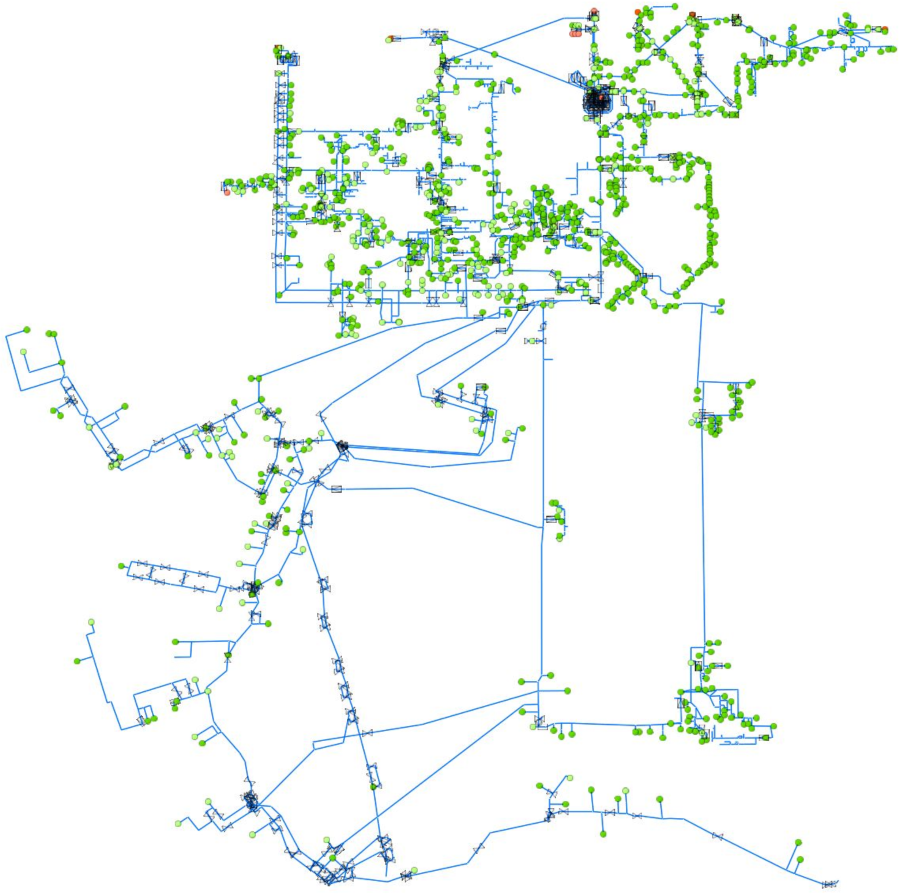
net	Baron (MINLP)	SCIP (MINLP)	MIP	NLP	MIP+NLP
1	<1s	<1s	<1s	<1s	<1s
2	<1s	<1s	<1s	<1s	<1s
3	456s	2s	2s	1s	3s
4	>1h	>1h	51.26s	30.83s	82.09s

L-Gas network of Open Grid Europe, Germany



Algebraic Model

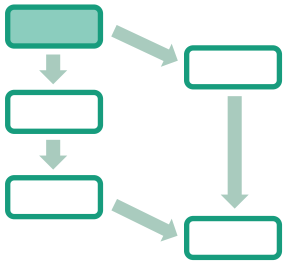
- 13 entries
- 1,062 exits
- 3,632 pipes
- 26 resistors
- 305 valves
- 120 control valve stations
- 12 compressor stations
- 25,000 variables (5,000 binary)



Computing time for 51 *expert scenarios*:

- 5 min to 70 min
- average: **34 min**

H-Gas network of Open Grid Europe, Germany

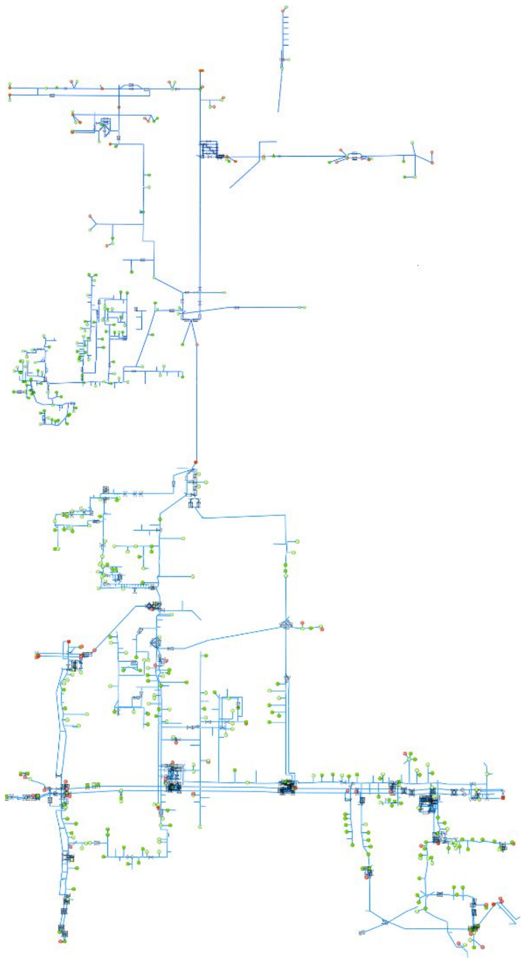


Algebraic Model

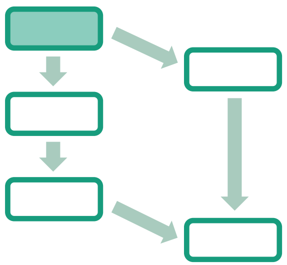
- 78 entries
- 395 exits
- 1,588 pipes
- 56 resistors
- 264 valves
- 101 control valve stations
- 35 compressor stations
- 35,000 variables (14,000 binary)

Computing time for 29 *expert scenarios*:

- 18 min to 10 hours
- average: **168 min**

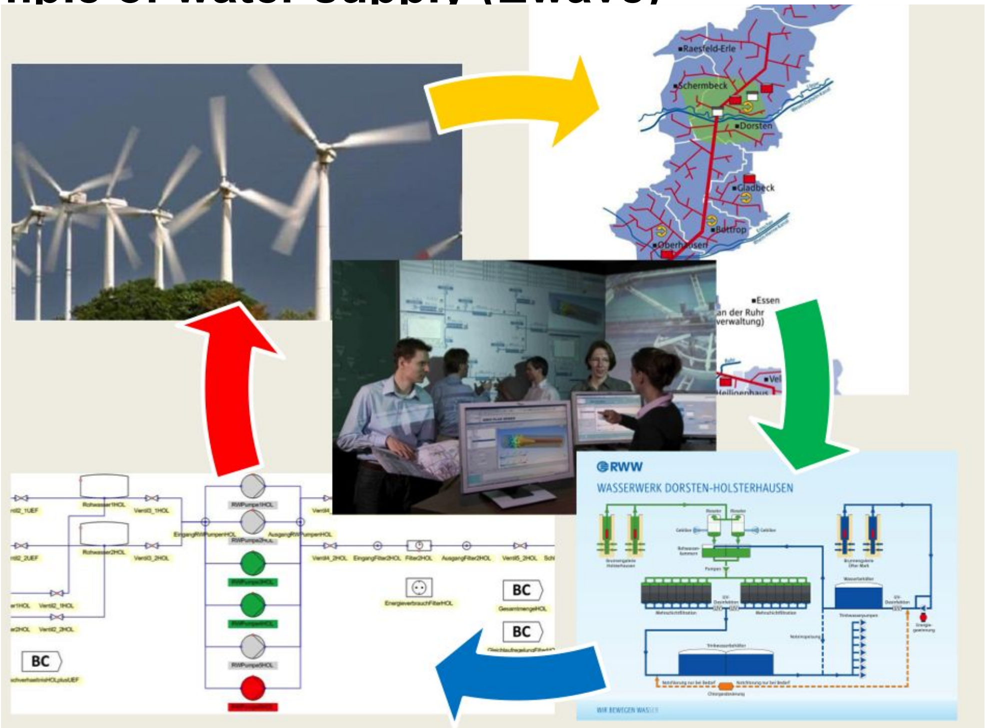


A Second Application: Energy Efficient Water Supply



Algebraic Model

Develop local energy management systems to improve energy optimal operating plans using the example of water supply (Ewave)



Partner:

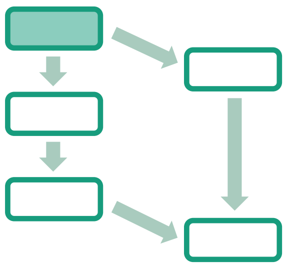
- FAU
- TU Darmstadt
- Uni Mannheim
- HS Bonn-Rhein-Sieg
- RWW
- Siemens

Federal Ministry of Education and Research

Friedrich-Alexander-Universität Erlangen-Nürnberg

Fraunhofer IIS

Pipe details – 'Water Hammer Equations'



Algebraic Model

Fundamental description by a system of hyperbolic partial differential equations

Continuity equation

$$\frac{\partial h}{\partial t} + \frac{a^2}{gA}\frac{\partial q}{\partial x} = 0$$

Momentum equation

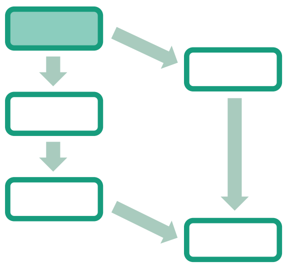
$$\frac{\partial q}{\partial t} + gA\frac{\partial h}{\partial x} = -\lambda\frac{q|q|}{2DA}$$

$$\frac{1}{\sqrt{\lambda}} = -2\log_{10}\left(\frac{k}{3.7D} + \frac{2.51}{\mathrm{Re}\sqrt{\lambda}}\right)$$

$$\mathrm{Re} = \frac{D}{\nu A}|q|$$

$h = h(x,t)$	(pressure) head
$q = q(x,t)$	flow
$\lambda = \lambda(q)$	friction factor
A	cross-sectional area
D	diameter
L	length

Pipe details – ’Water Hammer Equations’



Algebraic Model

After applying an implicit box scheme [Wendroff, 1960; Kolb, Lang, Bales, 2010]

Discretized continuity equation

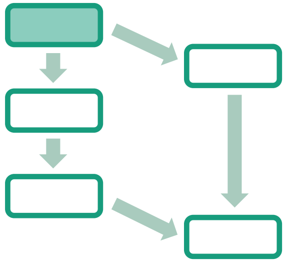
$$\frac{h_w^{t+1}+h_v^{t+1}}{2\Delta t}-\frac{h_w^{t}+h_v^{t}}{2\Delta t}+\frac{a^2}{gA}\frac{q_w^{t+1}-q_v^{t+1}}{L}=0$$

Discretized momentum equation

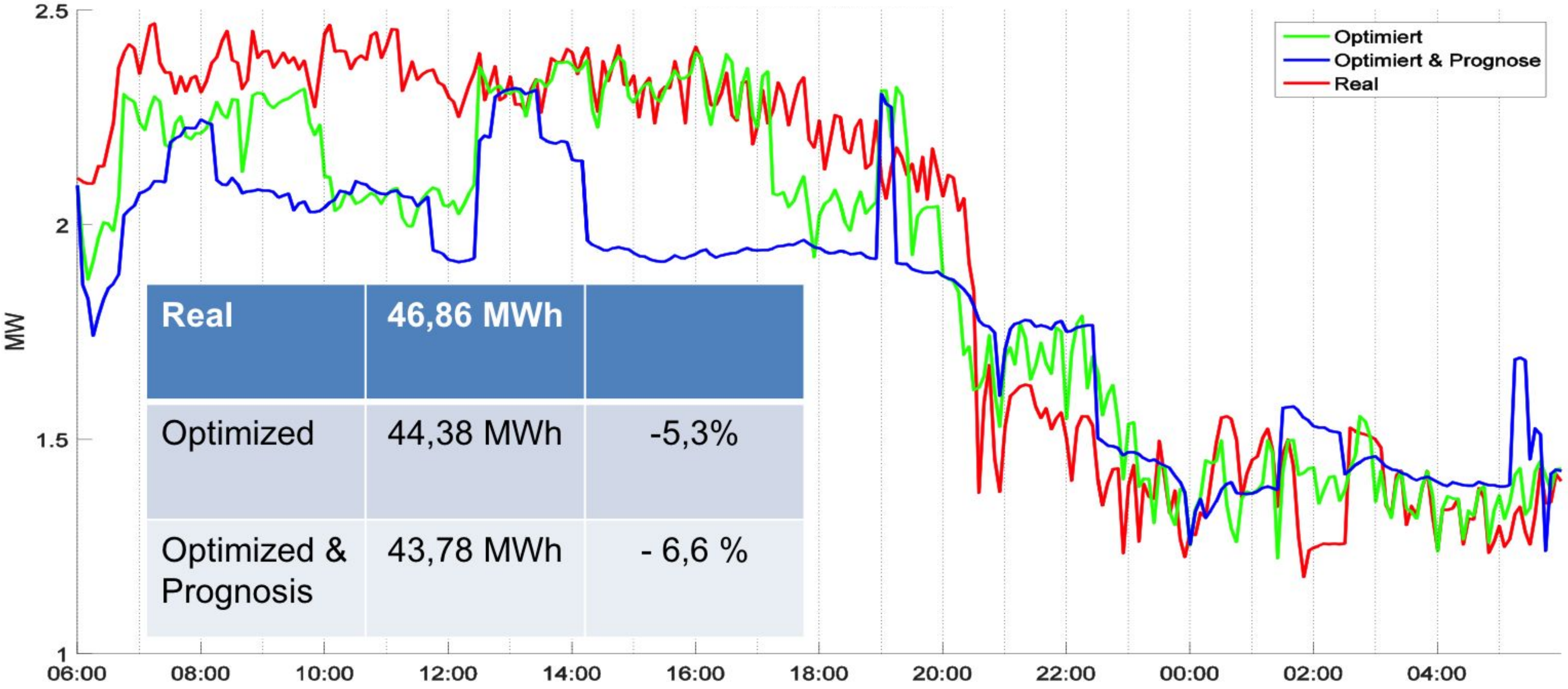
$$\frac{q_w^{t+1}+q_v^{t+1}}{2\Delta t}-\frac{q_w^{t}+q_v^{t}}{2\Delta t}+gA\frac{h_w^{t+1}-h_v^{t+1}}{L}=-\frac{1}{2DA}\left(\lambda_v^{t+1}\frac{q_v^{t+1}|q_v^{t+1}|}{2}+\lambda_w^{t+1}\frac{q_w^{t+1}|q_w^{t+1}|}{2}\right)$$

$h = h(x,t)$	(pressure) head
$q = q(x,t)$	flow
$\lambda = \lambda(q)$	friction factor
A	cross-sectional area
D	diameter
L	length

Numerical results

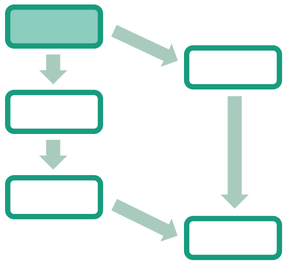


Algebraic Model

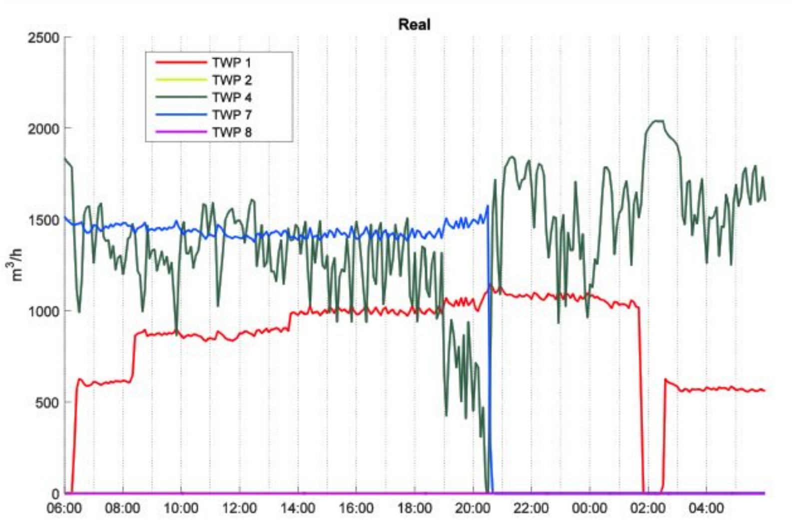


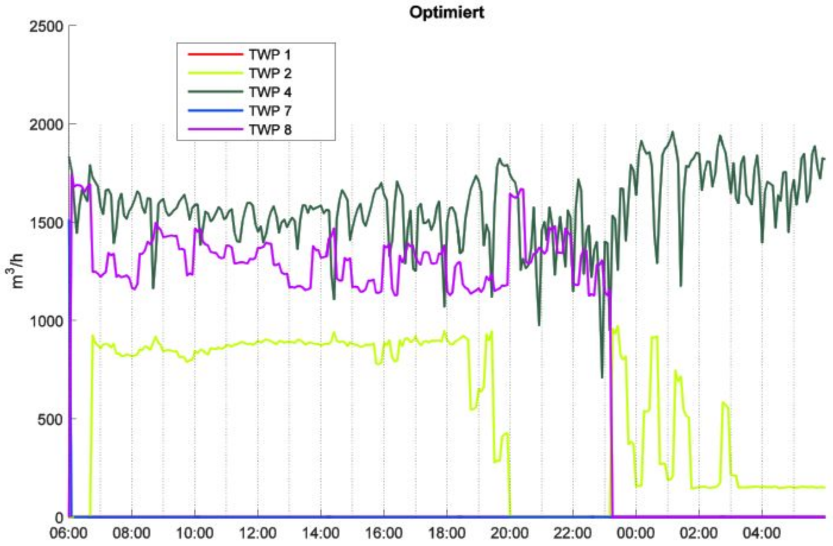
Real	46,86 MWh	
Optimized	44,38 MWh	-5,3%
Optimized & Prognosis	43,78 MWh	- 6,6 %

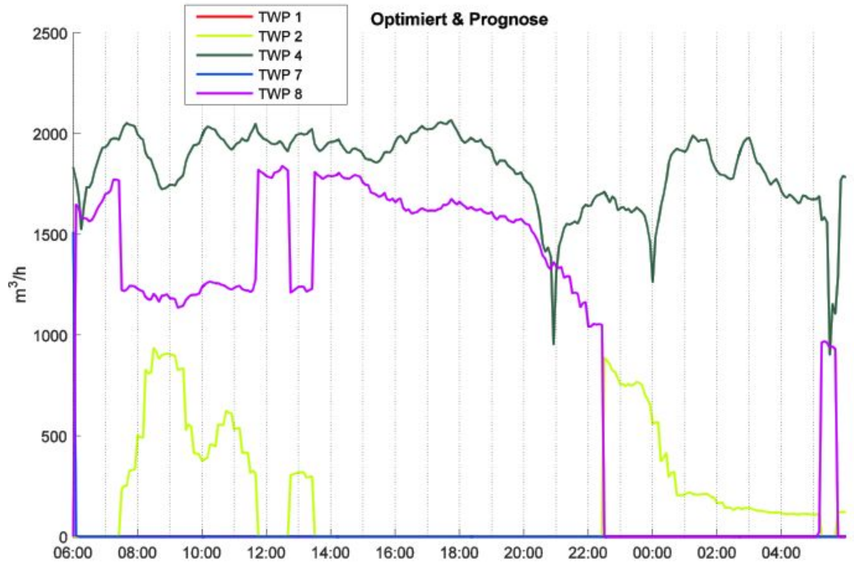
Optimized Pumping Schedule



Algebraic Model

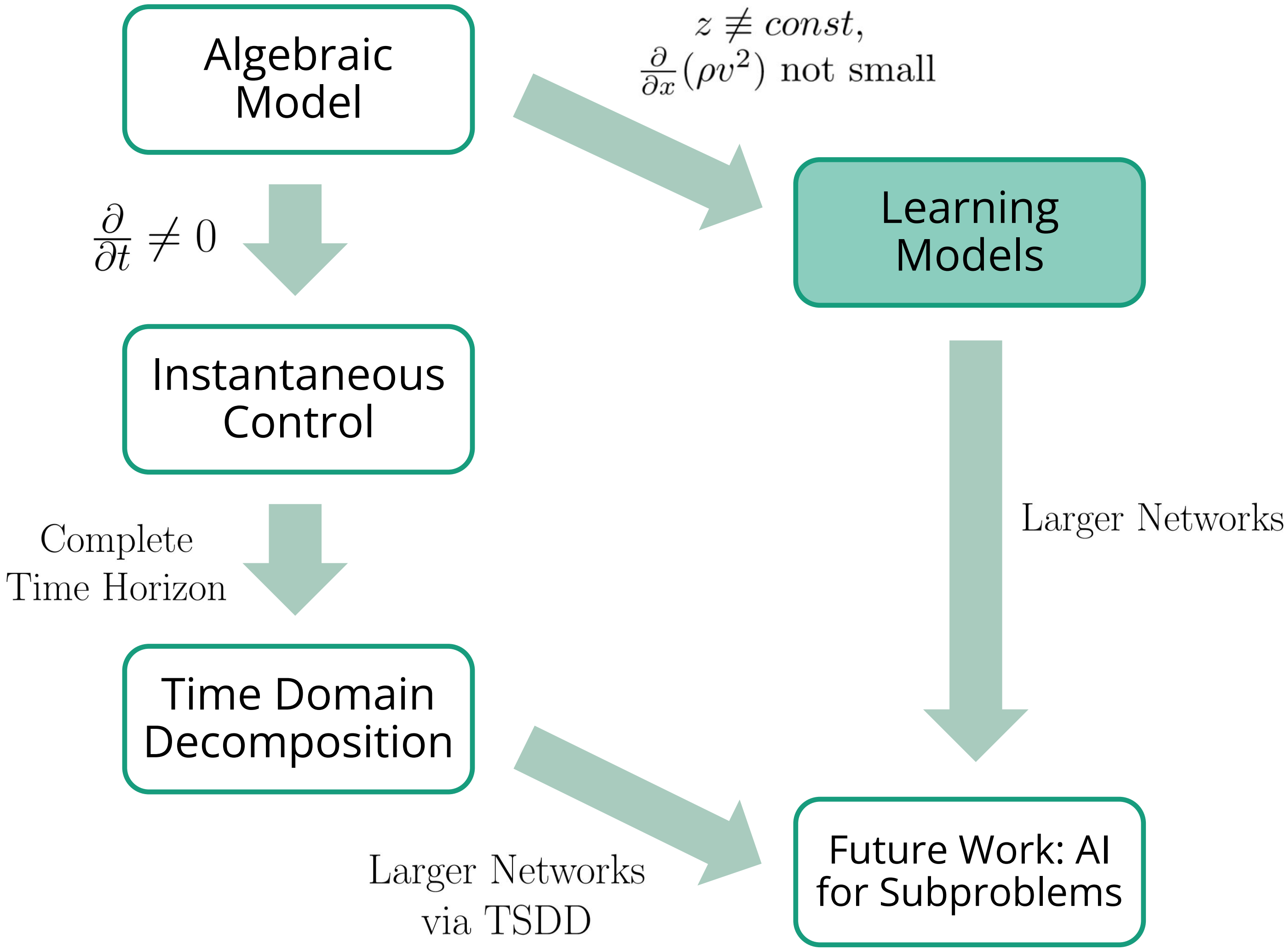




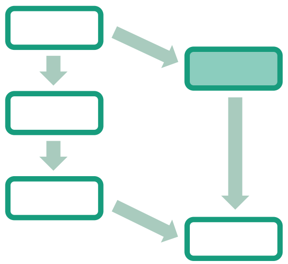


Friedrich-Alexander-Universität Erlangen-Nürnberg

Learning Model



Learning Model - learning instead of remodeling



Learning Models

Towards Simulation Based Mixed-Integer Optimization with Differential Equations

Gugat, Leugering, Martin, Schmidt, Sirvent, Wintergerst (2018)

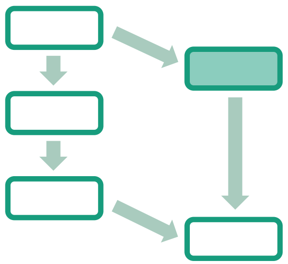
Definition (Masterproblem = MIP)

$$
\begin{aligned}
\min\ & c^T x + d^T y \\
\text{s.t. } & Ax + Bz \le b \\
& \text{linearize}(\mathcal{X}) \qquad\qquad \text{MIP}(\mathcal{X}) \\
& x \in [\underline{x}, \overline{x}],\ z \in [\underline{z}, \overline{z}] \\
& (x, z) \in \mathbb{R}^n \times \mathbb{Z}^m
\end{aligned}
$$

Definition (Subproblem PDE; ODE; DAE; ...)

$$
\begin{aligned}
& F(x, x_i, Dx_i, \ldots, D^{n-1}x_i) = 0 \qquad (\mathcal{X}) \\
& \forall i \in 1, \ldots, n
\end{aligned}
$$

Learning Model - learning instead of remodeling



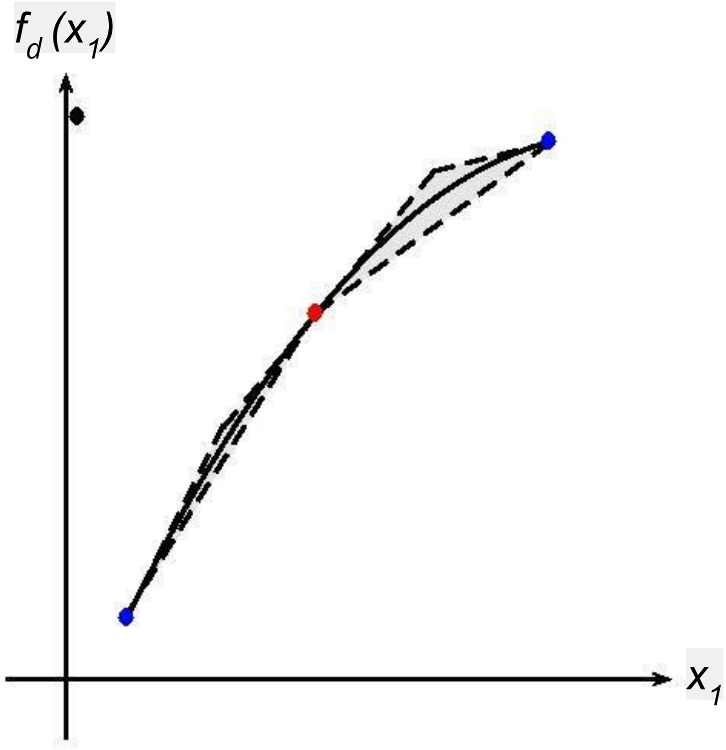
Learning Models

Definition (Masterproblem = MIP)

$$\begin{aligned} &\min c^T x + d^T y \\ &\text{s.t. } Ax + Bz \leq b \\ &\quad \text{linearize}(\mathcal{X}) \qquad\qquad \text{MIP}(\mathcal{X}) \\ &\quad x \in [\underline{x}, \overline{x}],\ z \in [\underline{z}, \overline{z}] \\ &\quad (x, z) \in \mathbb{R}^n \times \mathbb{Z}^m \end{aligned}$$

Definition (Subproblem PDE; ODE; DAE; ...)

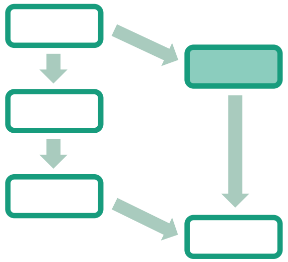
$$\begin{aligned} &F(x, x_i, Dx_i, \dots, D^{n-1}x_i) = 0 \qquad (\mathcal{X}) \\ &\forall i \in 1, \dots, n \end{aligned}$$



$\mathcal{X} = \{\bullet, \bullet, \bullet\}$

TRR 154 UTN FAU Friedrich-Alexander-Universität Erlangen-Nürnberg Fraunhofer IIS

Learning Model - The 1-dimensional case



Learning Models

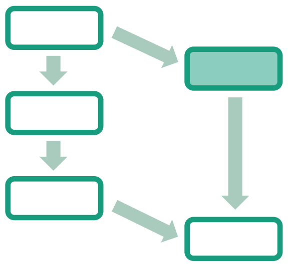
Assumption

The functions f_d are

- not known explicitly
- strictly monotonic
- strictly concave or convex
- differentiable with bounded first derivative

$$
\begin{aligned}
\min_{x} \quad & c^T x \\
\text{s.t.} \quad & Ax \geq b \\
& x_{d_2} = f_d(x_{d_1}) \qquad \text{for all } d \in \mathbb{D} \\
& \hat{x} \leq x \leq \bar{x} \\
& x_{\mathbb{C}} \in \mathbb{R}^{|\mathbb{C}|},\ x_{\mathcal{I}} \in \mathbb{Z}^{|\mathcal{I}|}
\end{aligned}
$$

Learning Model - The 1-dimensional case



Learning Models

Theorem

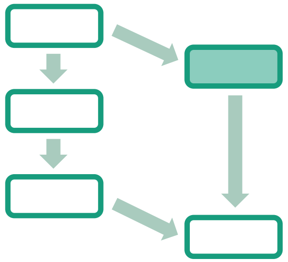
- Ordinary differential equation

  $$y' = g(x, y(x)), \quad y(0) = y_0, \quad x \in [0, L]$$

- Solution $y = y(x; y_0)$
- f maps y_0 onto solution of initial value problem, i.e., $f(y_0) = y(L; y_0)$

$$
\begin{aligned}
\min_x \quad & c^T x \\
\text{s.t.} \quad & Ax \geq b \\
& x_{d_2} = f_d(x_{d_1}) \quad \text{for all } d \in \mathbb{D} \\
& \hat{x} \leq x \leq \bar{x} \\
& x_{\mathbb{C}} \in \mathbb{R}^{|\mathbb{C}|}, \ x_{\mathcal{I}} \in \mathbb{Z}^{|\mathcal{I}|}
\end{aligned}
$$

Learning Model - The 1-dimensional case



Learning Models

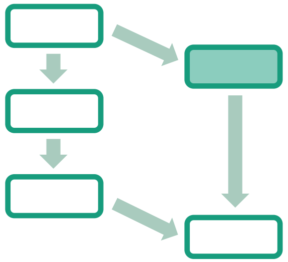
Theorem (Gugat, Leugering, Martin, Schmidt, Sirvent, Wintergerst 2018)

An algorithm based on iteratively cutting off the relaxed solution terminates after a finite number of steps with

- an ε-feasible solution *or*
- an indication of infeasibility

$$
\begin{aligned}
\min_{x} \quad & c^T x \\
\text{s.t.} \quad & Ax \geq b \\
& x_{d_2} = f_d(x_{d_1}) \quad \text{for all } d \in \mathbb{D} \\
& \hat{x} \leq x \leq \bar{x} \\
& x_{\mathbb{C}} \in \mathbb{R}^{|\mathbb{C}|}, \; x_{\mathcal{I}} \in \mathbb{Z}^{|\mathcal{I}|}
\end{aligned}
$$

Application: Stationary gas transport optimization

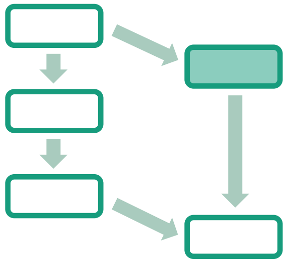


Learning Models

Greek Natural Gas Transport Network

Type	Quantity
Entries	3
Exits	45
Inner nodes	86
Pipes	86
Short pipes	45
Control valves	1
Compressors	1

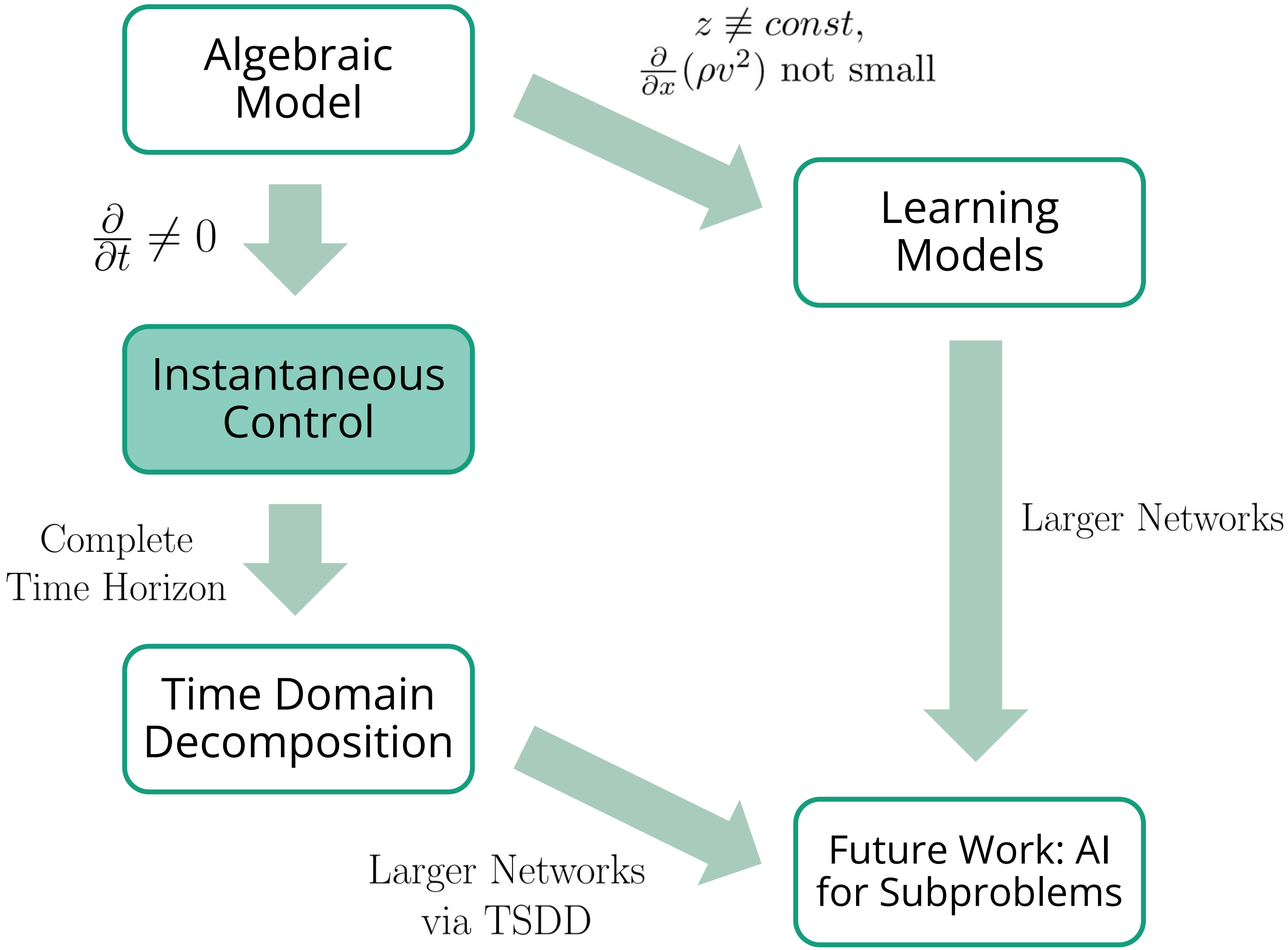
Application: Stationary gas transport optimization



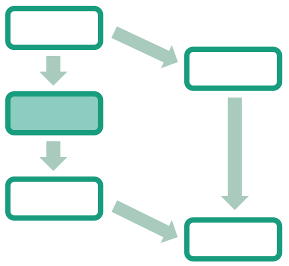
Learning Models

Instance	Status	Obj.	k	Total	Master	Sub
12/23/2011	opt.	262.91	6	29.15	0.13	29.03
04/19/2012	opt.	0	5	23.50	0.14	23.36
10/08/2012	opt.	248.96	6	27.30	0.22	27.09
03/16/2013	inf.	—	2	7.75	0.01	7.74
01/25/2014	opt.	311.4	5	23.73	0.09	23.64
07/04/2014	opt.	335.23	5	23.01	0.06	22.95
09/07/2014	opt.	0	6	26.55	0.38	26.17
11/14/2014	opt.	0	6	33.52	0.30	33.22
08/27/2015	opt.	0	5	22.90	0.14	22.76
11/06/2015	inf.	—	2	8.43	0.01	8.42

Instantaneous Control



Instantaneous Control



Instantaneous Control

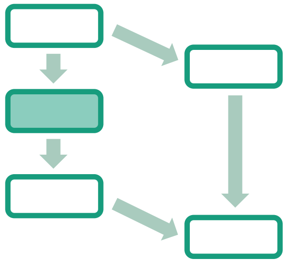
MIP-based Instantaneous Control Algorithm

for $\kappa = 0, \ldots, K-1$ **do**
    Setup optimization problem $(\mathrm{P}^{\kappa+1})$ for time step $t_{\kappa+1}$ that only depends on the state at t_κ and solve $(\mathrm{P}^{\kappa+1})$

with

$$
\begin{aligned}
\min_{x^{t_{\kappa+1}}} \quad & \left(c^{t_{\kappa+1}}\right)^\top x^{t_{\kappa+1}} \\
\text{s.t.} \quad & A_\kappa^{t_{\kappa+1}} x^{t_{\kappa+1}} \geq b_\kappa^{t_{\kappa+1}} \\
& x^- \leq x^{t_{\kappa+1}} \leq x^+ \\
& x_{\mathcal{C}}^{t_{\kappa+1}} \in \mathbb{R}^{|\mathcal{C}|}, \quad x_{\mathcal{I}}^{t_{\kappa+1}} \in \mathbb{Z}^{|\mathcal{I}|}
\end{aligned}
\qquad (\mathrm{P}^{\kappa+1})
$$

Instantaneous Control - Derivation of the Algorithm



Instantaneous Control

Isothermal nonlinear Euler equations

$$\partial_t p + \frac{R_s T z(p)}{A_a} \partial_x q = 0$$

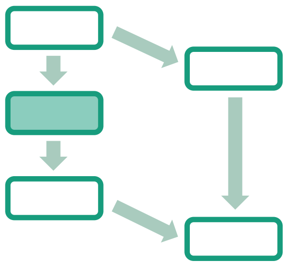
$$\partial_t \frac{q}{A_a} + \partial_x \left(p + \frac{R_s T z(p)}{A_a^2} \frac{q^2}{p} \right) = -\frac{\theta_a R_s T z(p)}{2 A_a^2} \frac{q|q|}{p} - \frac{g\, s_a}{R_s T z(p)} p$$

Semilinear Euler equations

$$\partial_t p + \frac{c^2}{A_a} \partial_x q = 0$$

$$\partial_t \frac{q}{A_a} + \partial_x p = -\frac{\theta_a c^2}{2 A_a^2} \frac{q|q|}{p} - \frac{g s_a}{c^2} p$$

Instantaneous Control - Derivation of the Algorithm



Instantaneous Control

Isothermal nonlinear Euler equations

$$\partial_t p + \frac{R_s T z(p)}{A_a} \partial_x q = 0$$

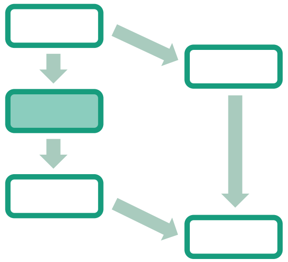
$$\partial_t \frac{q}{A_a} + \partial_x \left(p + \frac{R_s T z(p)}{A_a^2} \frac{q^2}{p} \right) = -\frac{\theta_a R_s T z(p)}{2 A_a^2} \frac{q|q|}{p} - \frac{g\, s_a}{R_s T z(p)} p$$

Semilinear Euler equations

$$\partial_t p + \frac{c^2}{A_a} \partial_x q = 0$$

$$\partial_t \frac{q}{A_a} + \partial_x p = -\frac{\theta_a c^2}{2 A_a^2} \frac{q|q|}{p} - \frac{g s_a}{c^2} p$$

Instantaneous Control - Derivation of the Algorithm
Discretizations



Instantaneous Control

Mixed implicit-explicit Euler discretization with time steps Δt_κ yields the ODE

$$\frac{p_{\kappa+1} - p_\kappa}{\Delta t_\kappa} + \frac{c^2}{A_a}\partial_x q_{\kappa+1} = 0$$

$$\frac{q_{\kappa+1} - q_\kappa}{\Delta t_\kappa A_a} + \partial_x p_{\kappa+1} = -\frac{\theta_a c^2}{2A_a^2}\frac{|q_\kappa| q_\kappa}{p_\kappa} - \frac{g s_a}{c^2} p_\kappa \qquad (1)$$

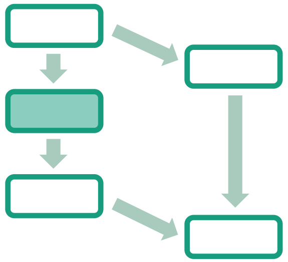
Theorem (Gugat, Leugering, Martin, Schmidt, Sirvent, Wintergerst 2017)

The solution of the ODE system (1) satisfies

$$q_{\kappa+1}(x) = \xi_1(x, \Delta t_\kappa) q_{\kappa+1}(0) + \xi_2(x, \Delta t_\kappa) p_{\kappa+1}(0) + l_1(x, q_\kappa(x), p_\kappa(x)),$$

$$p_{\kappa+1}(x) = \xi_3(x, \Delta t_\kappa) q_{\kappa+1}(0) + \xi_4(x, \Delta t_\kappa) p_{\kappa+1}(0) + l_2(x, q_\kappa(x), p_\kappa(x)).$$

Instantaneous Control - Derivation of the Algorithm
Discretizations



Instantaneous Control

Mixed implicit-explicit Euler discretization with time steps Δt_κ yields the ODE

$$\begin{aligned} \frac{p_{\kappa+1} - p_\kappa}{\Delta t_\kappa} + \frac{c^2}{A_a}\partial_x q_{\kappa+1} &= 0 \\ \frac{q_{\kappa+1} - q_\kappa}{\Delta t_\kappa A_a} + \partial_x p_{\kappa+1} &= -\frac{\theta_a c^2}{2A_a^2}\frac{|q_\kappa| q_\kappa}{p_\kappa} - \frac{g s_a}{c^2} p_\kappa \end{aligned} \tag{1}$$

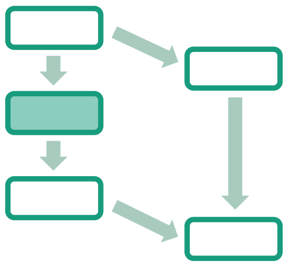
Theorem (Gugat, Leugering, Martin, Schmidt, Sirvent, Wintergerst 2017)

The solution of the ODE system (1) satisfies

$$q_{\kappa+1}(x) = \xi_1(x, \Delta t_\kappa) q_{\kappa+1}(0) + \xi_2(x, \Delta t_\kappa) p_{\kappa+1}(0) + l_1(x, q_\kappa(x), p_\kappa(x)),$$

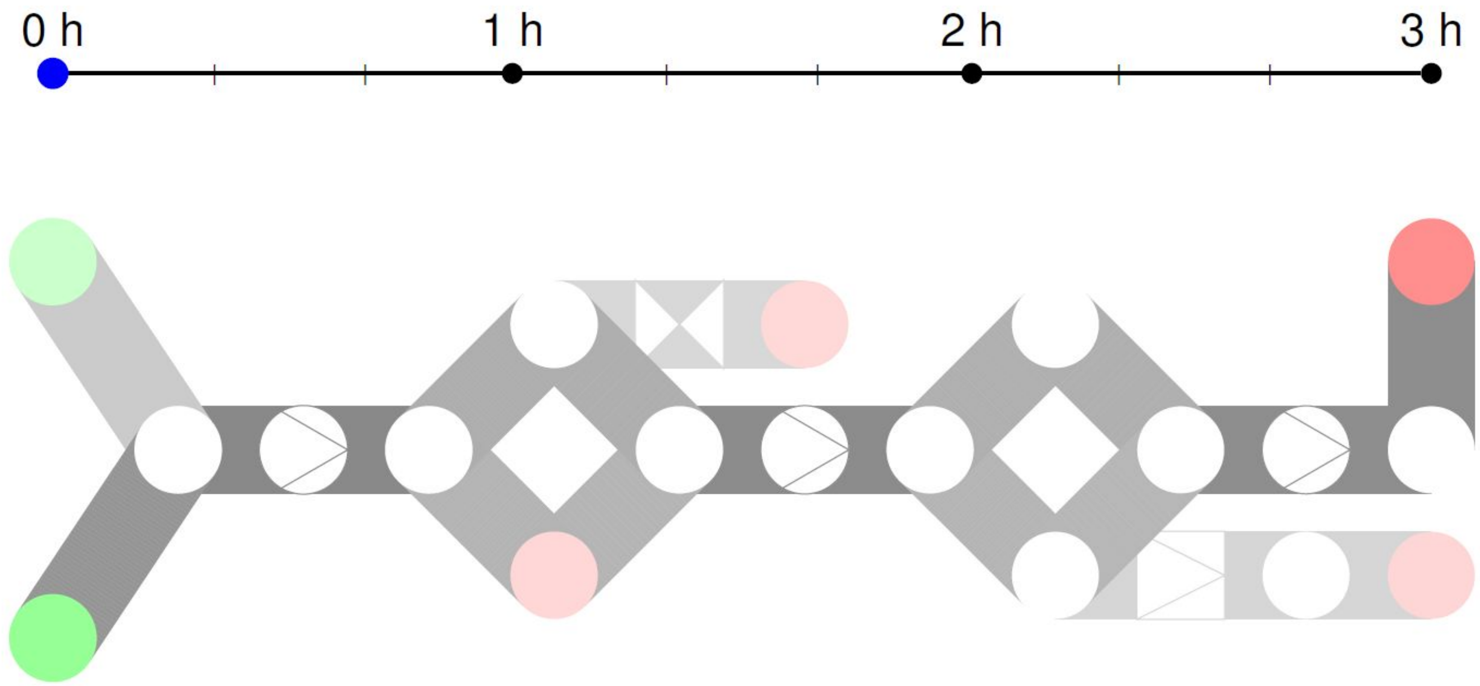
$$p_{\kappa+1}(x) = \xi_3(x, \Delta t_\kappa) q_{\kappa+1}(0) + \xi_4(x, \Delta t_\kappa) p_{\kappa+1}(0) + l_2(x, q_\kappa(x), p_\kappa(x)).$$

Instantaneous Control – Numerical results

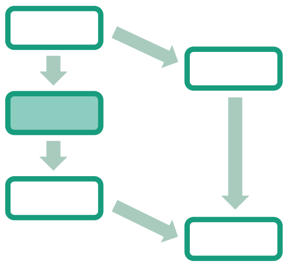


Instantaneous Control

Transient Gas Transport Optimization – Start

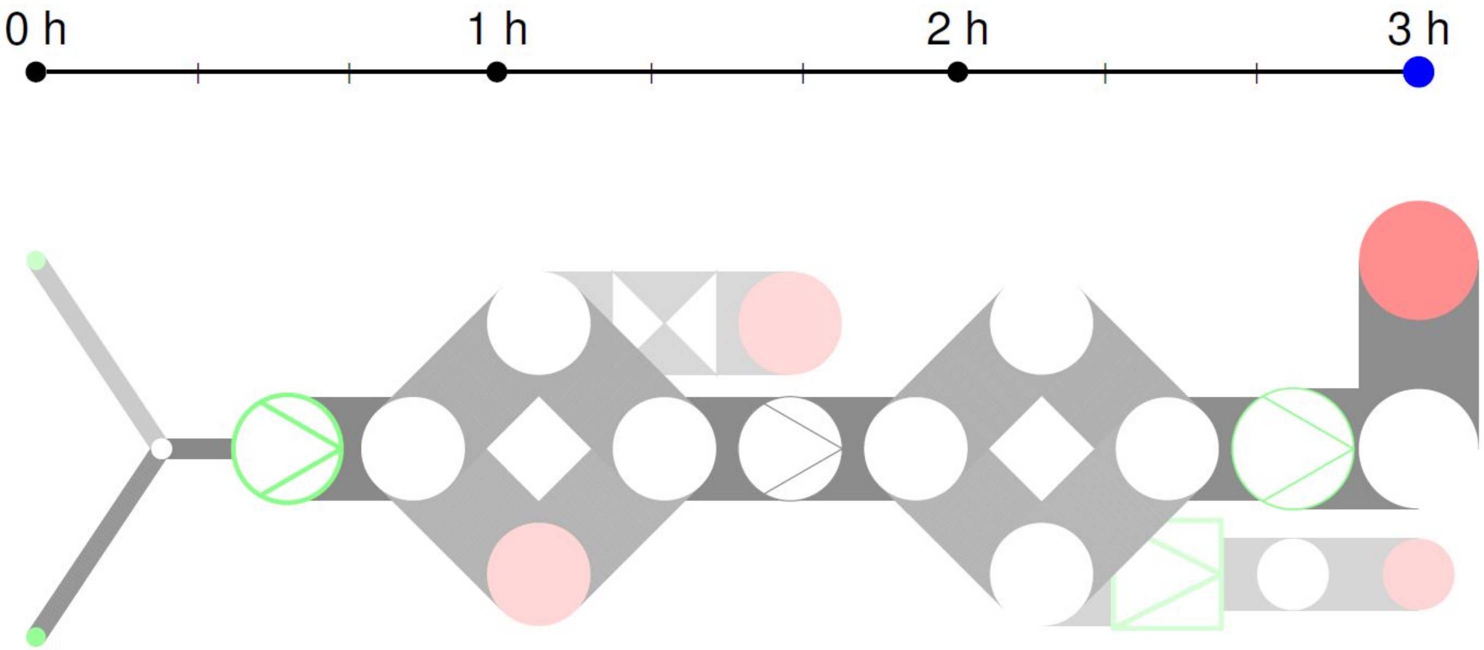


Instantaneous Control – Numerical results

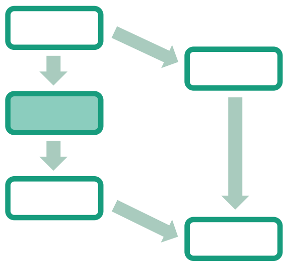


Instantaneous Control

Transient Gas Transport Optimization – End

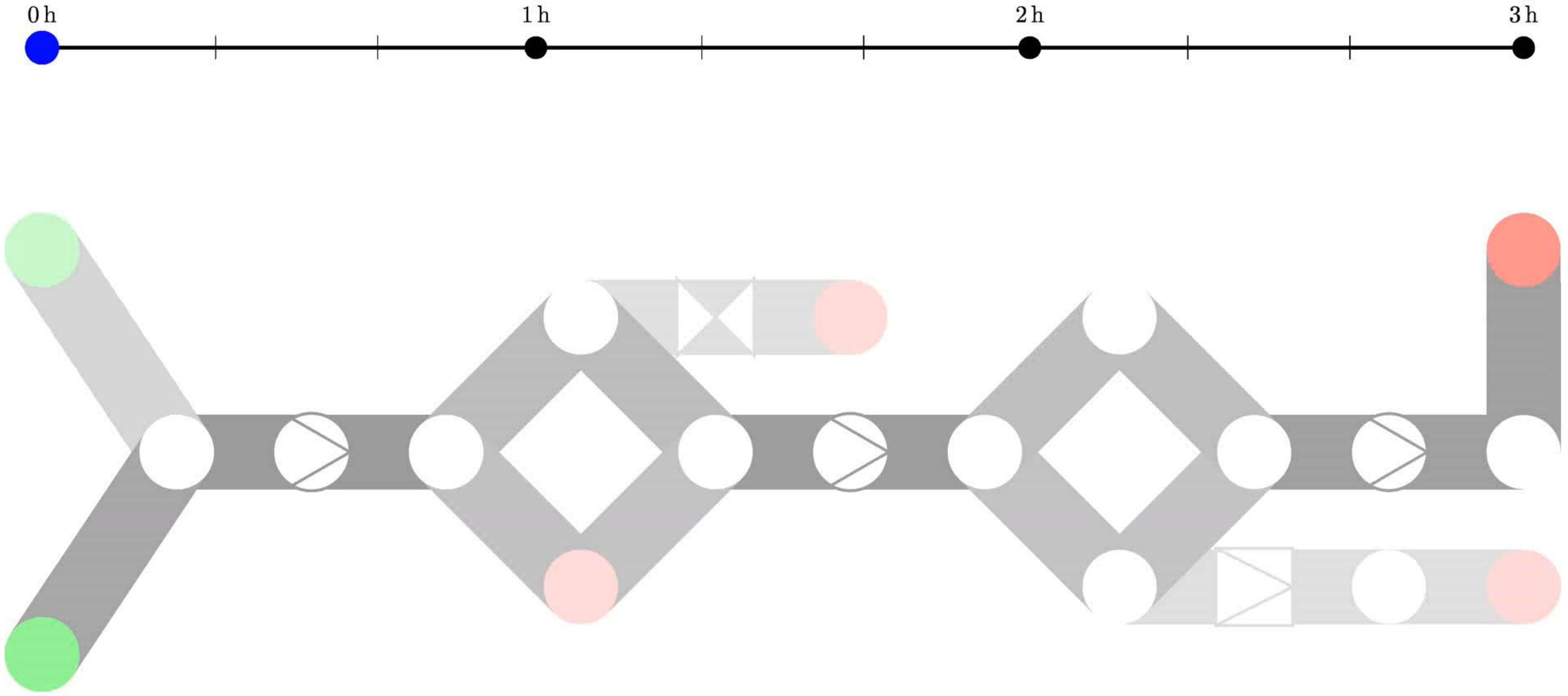


Instantaneous Control – Numerical results

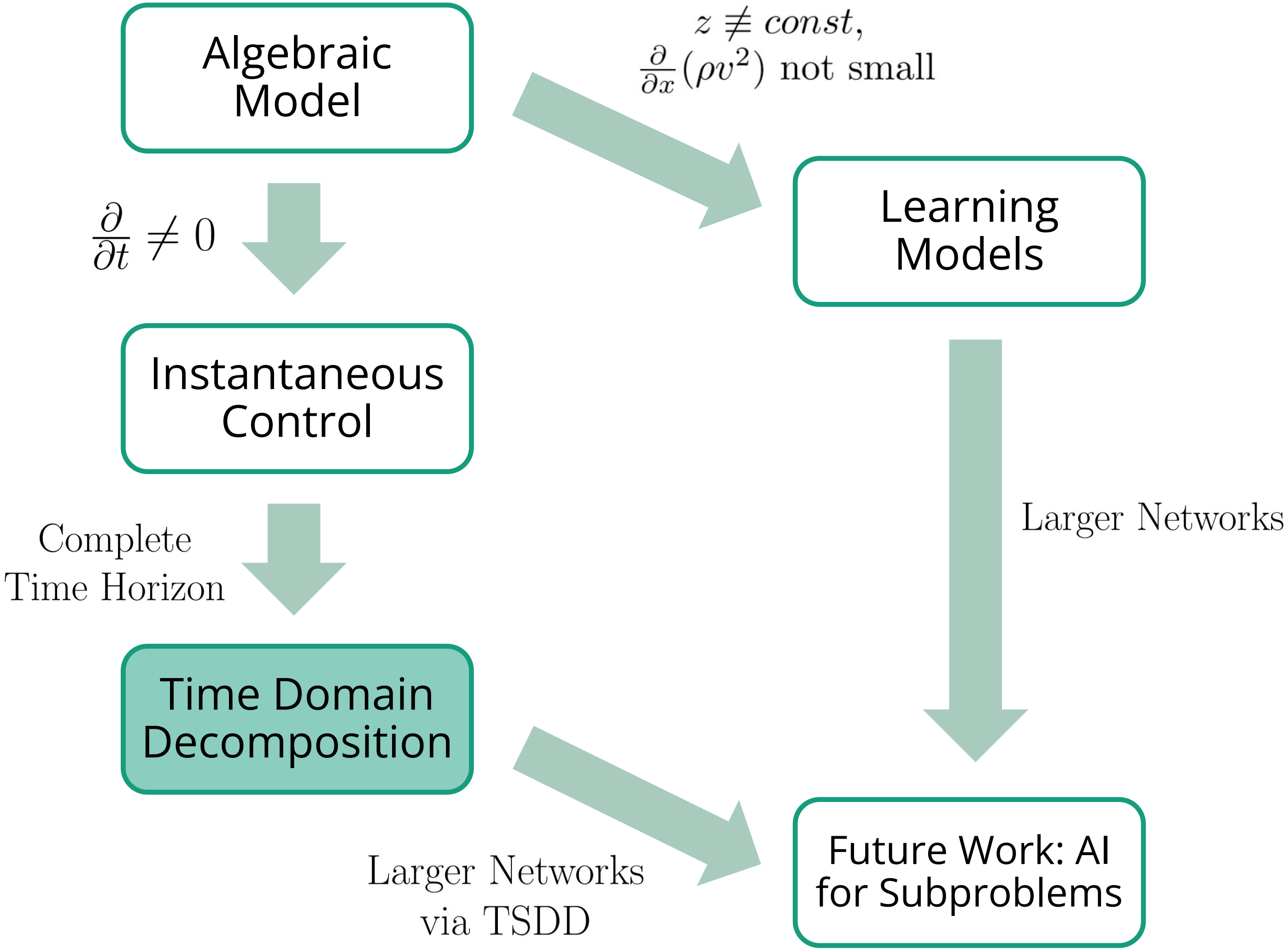


Instantaneous Control

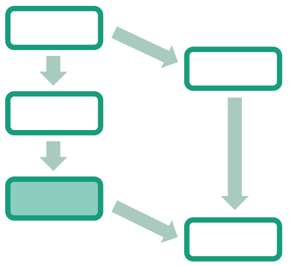
Transient Gas Transport Optimization – End



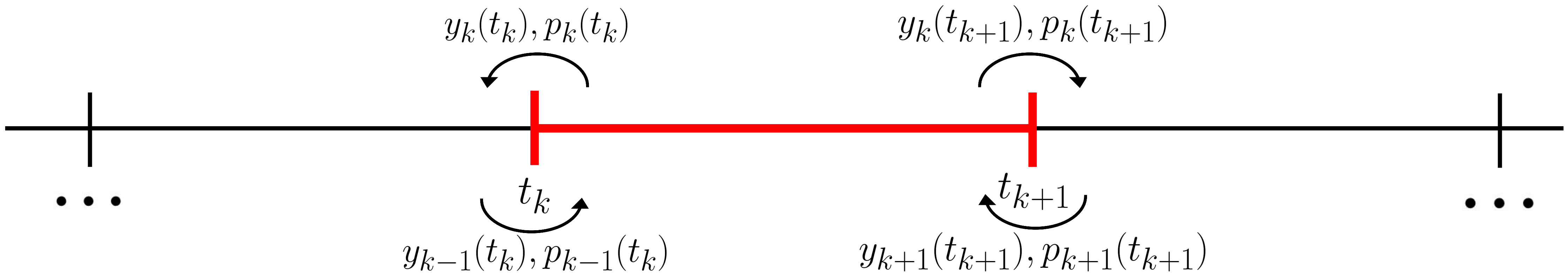
Time Domain Decomposition



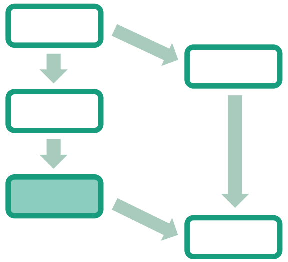
Time Domain Decomposition - Concept



- Set up optimality system of the overall problem
- Non-overlapping time-domain decomposition of this optimality system
- Decouple sub-domain systems using an iterative update rule
- All sub-domain problems can be solved in parallel
- Sub-domain problems have a primal interpretation as so-called virtual control problems



Time Domain Decomposition – Proofs and Insights



Time Domain Decomposition

- We provide a proof of convergence and a posteriori error estimates

  R. Krug, G. Leugering, A. Martin, M. Schmidt, D. Weninger. Time-Domain Decomposition for Optimal Control Problems Governed by Semilinear Hyperbolic Systems. SIAM Journal on Control and Optimization, 2021.

- We extend the approach to problems with mixed two-point boundary conditions

  R. Krug, G. Leugering, A. Martin, M. Schmidt, D. Weninger. Time-Domain Decomposition for Optimal Control Problems Governed by Semilinear Hyperbolic Systems with Mixed Two-Point Boundary Conditions. Control and Cybernetics, 2021.

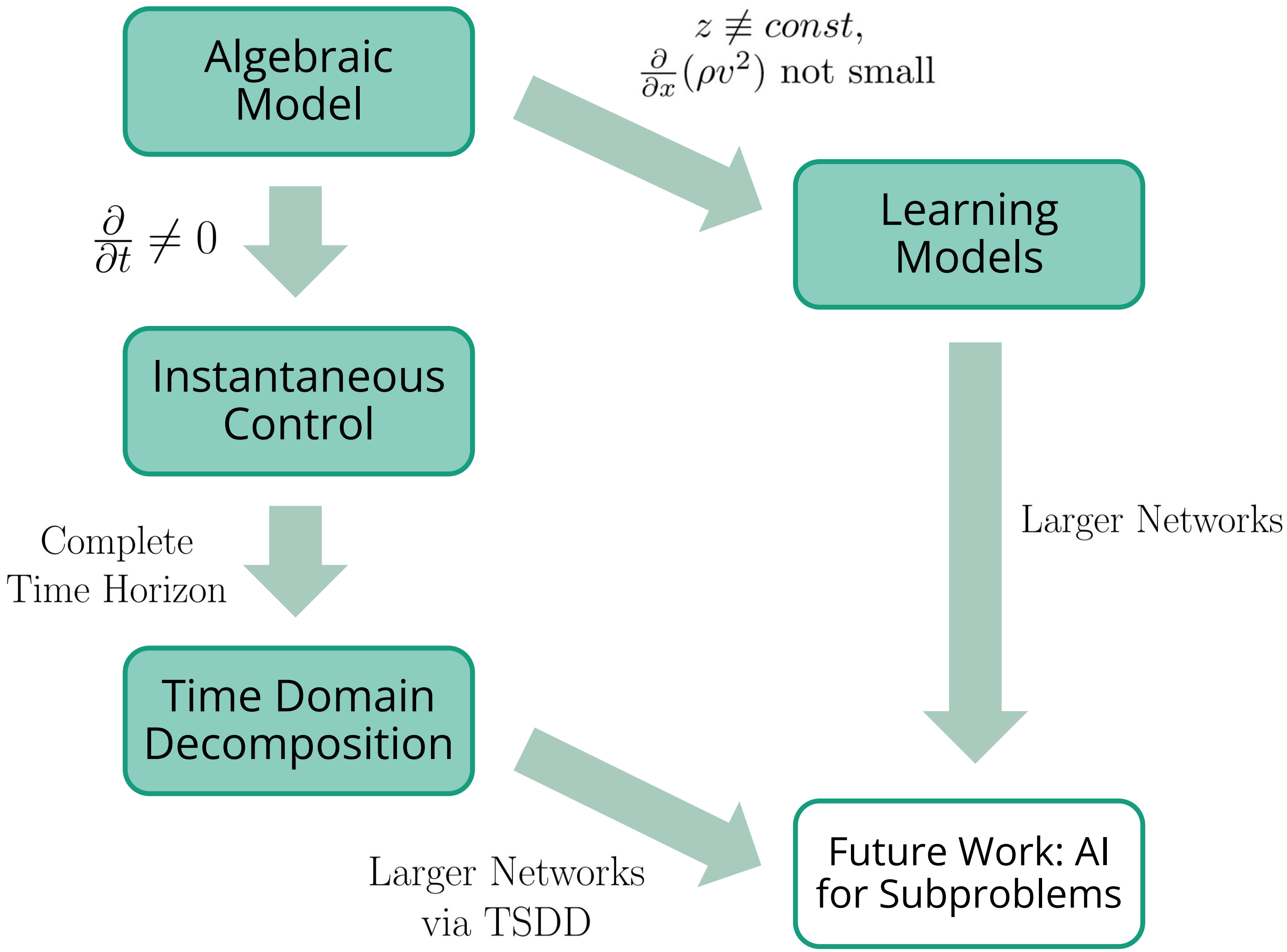
- A combined simultaneous space-time-domain decomposition is tackled

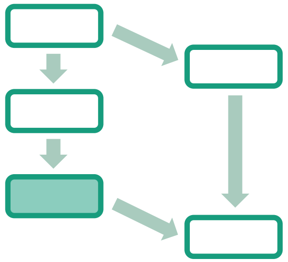
  G. Leugering. Space-Time-Domain Decomposition for Optimal Control Problems Governed by Linear Hyperbolic Systems. Journal of Optimization, Differential Equations and Their Applications, 2021.

- A time-domain decomposition for ODE-constrained MINLPs is given

  F. M. Hante, R. Krug, M. Schmidt. Time-Domain Decomposition for Mixed-Integer Optimal Control Problems. Preprint, TRR 154, 2021.

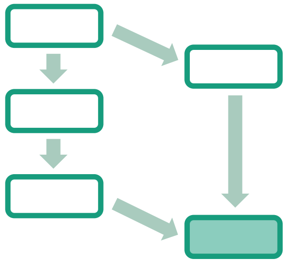
So far…





Time Domain Decomposition

Where we are – Where to go



Future Work

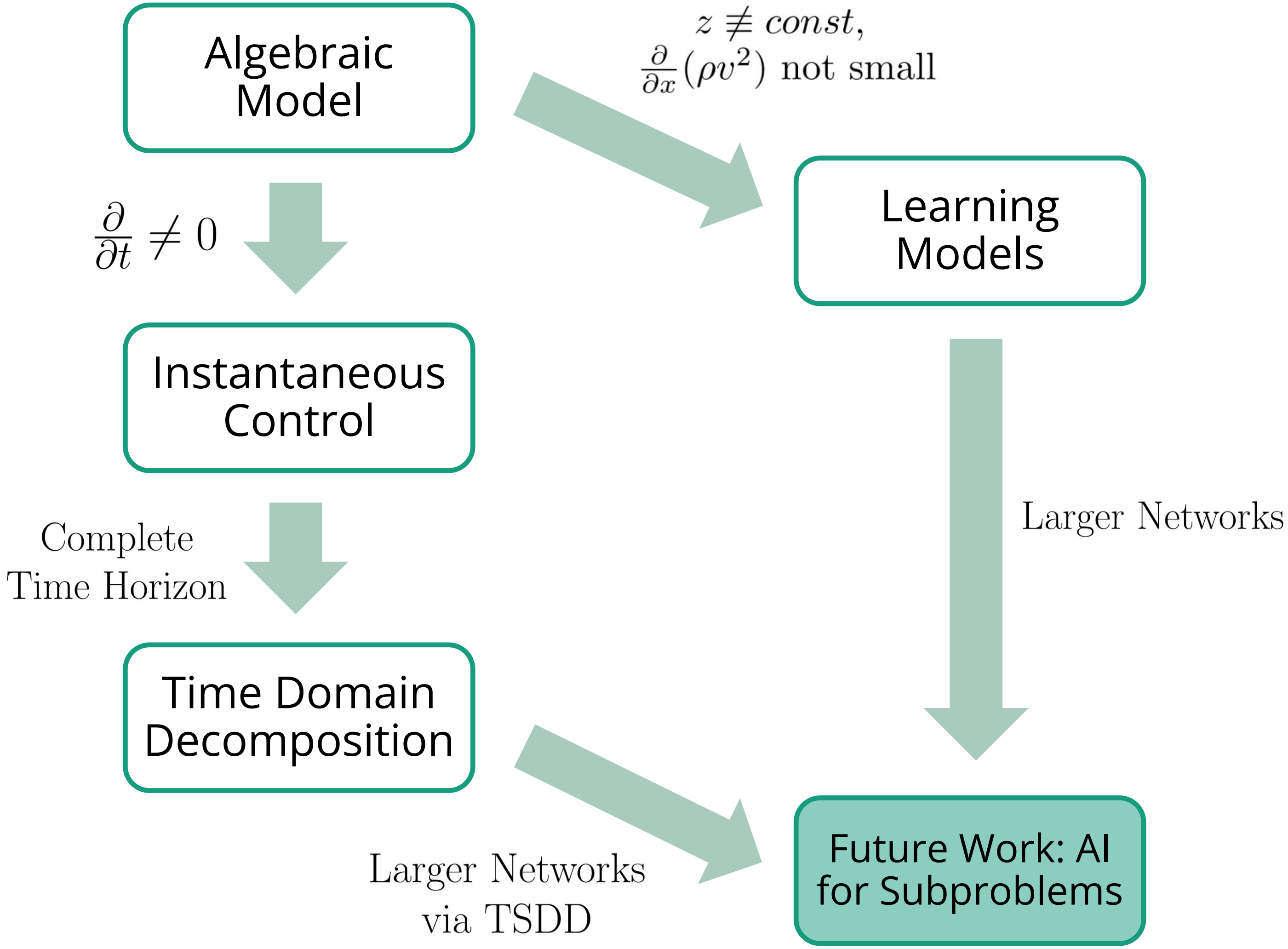
Where we are

- Stationary case is solvable
- The non-steady case
  - Instantaneous control
  - Decomposition approaches with limits

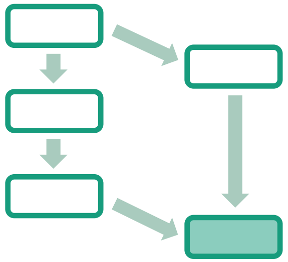
Where to go

- Remodeling and analytic approaches will stay an important topic
- Further need for consistent hierarchy of all models
- Including techniques from machine learning

Future Work


Algebraic Model
$z \not\equiv const,$
$\frac{\partial}{\partial x}(\rho v^2)$ not small
Learning Models
$\frac{\partial}{\partial t} \neq 0$
Instantaneous Control
Complete Time Horizon
Time Domain Decomposition
Larger Networks
Larger Networks via TSDD
Future Work: AI for Subproblems

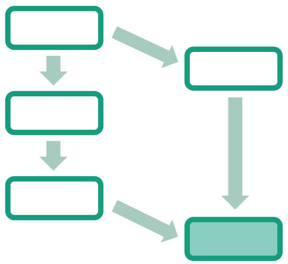
Future Work - Goals and mathematical challenges



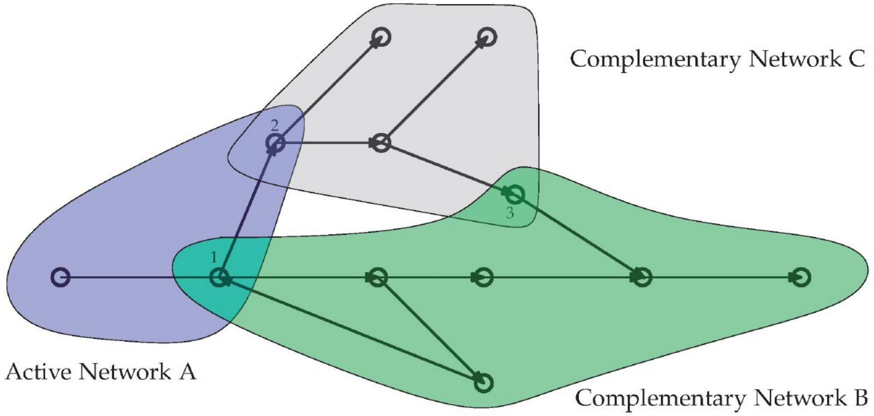
Future Work

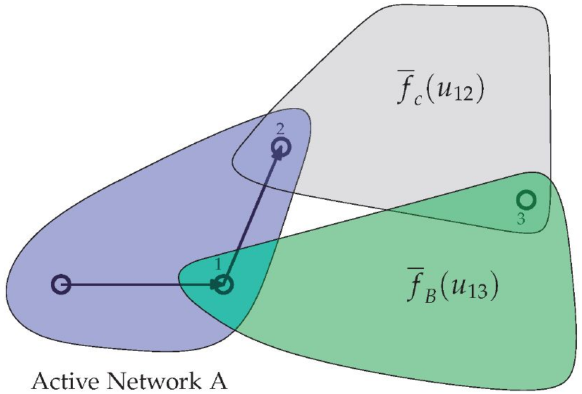
- Couple the space-time-domain decomposition methods with machine learning and mixed-integer programming techniques
- Development of an interlinked data-driven and physics-informed algorithm called NeTI (Network Tearing and Interconnection)
- NeTI combines
  - Mixed-integer nonlinear programming
  - Learning of surrogate models
  - Graph decomposition strategies

Future Work – Network Tearing and Interconnection

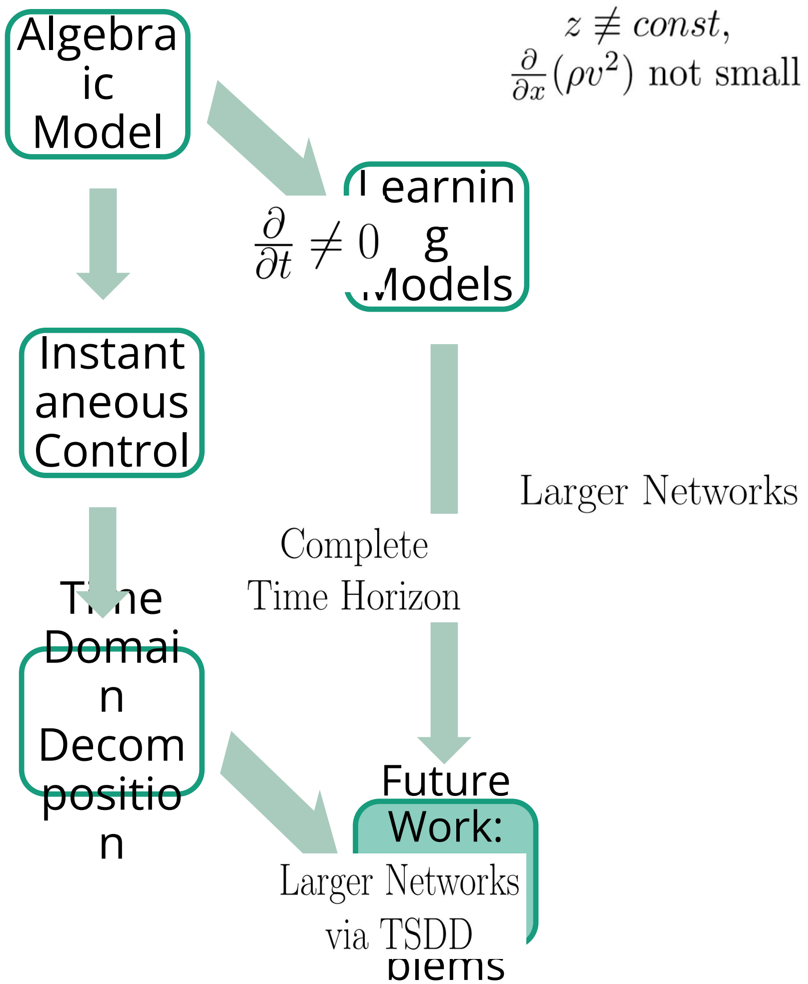


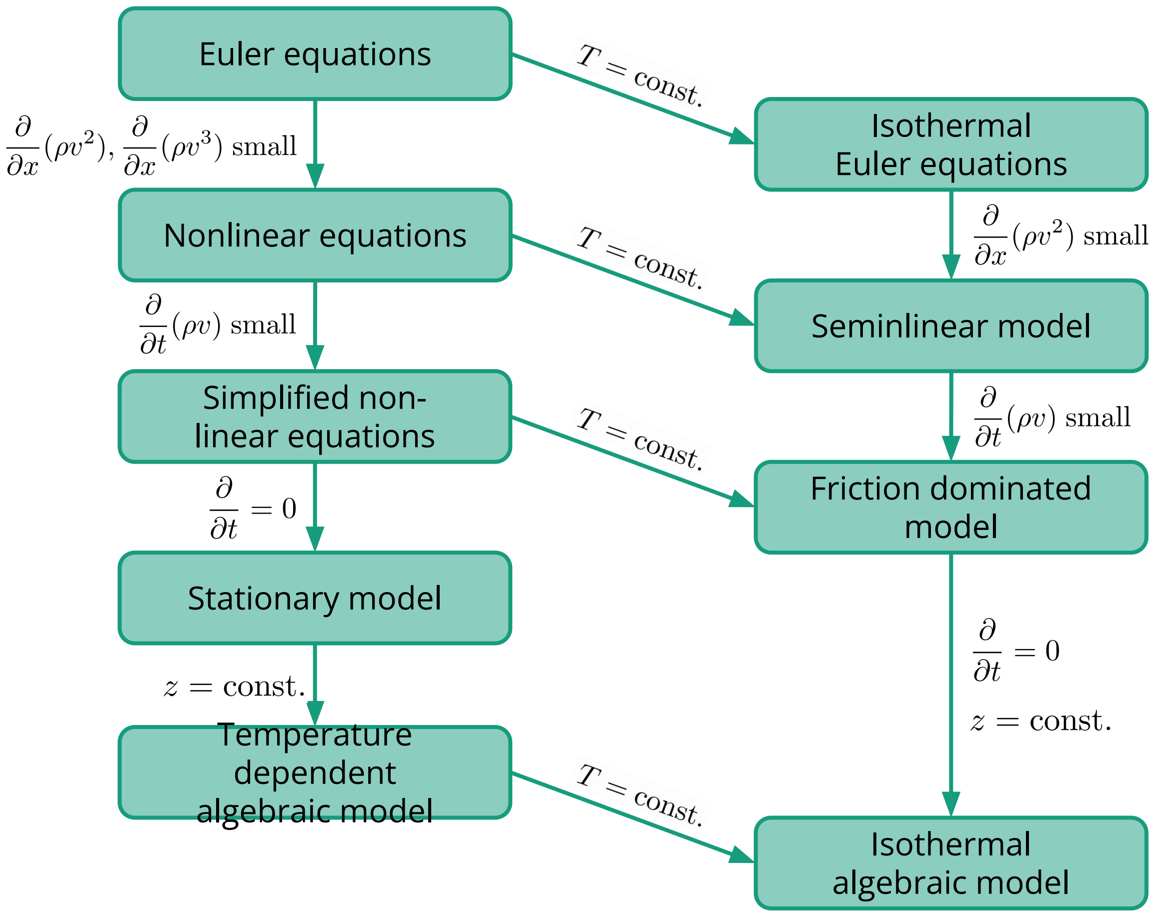
- Network Tearing and Interconnection (NeTI)
  - Choose the active network and decompose the complement graph into sub-graphs (Tearing)
  - Learn surrogate dynamics of all sub-graphs
  - Solve the MINLP on the active graph using surrogates at the interfaces with penalties to ensure continuity
  - Penalty update (Interconnection)



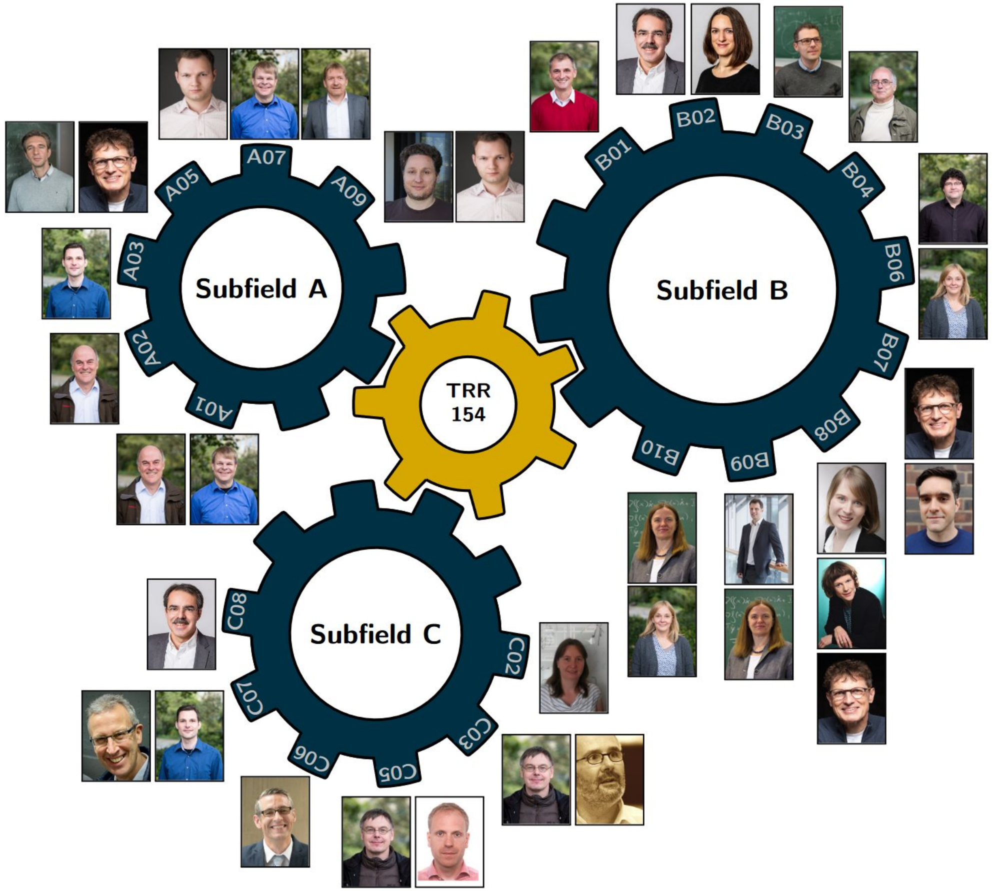


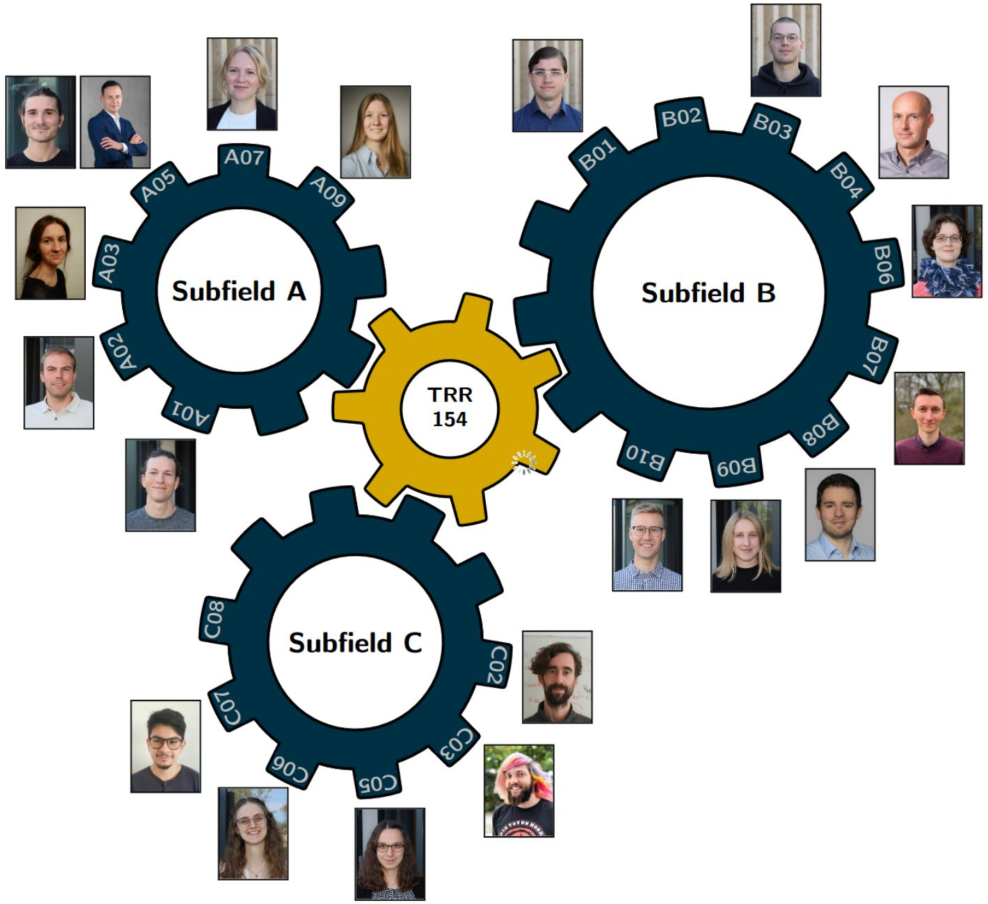
Future Work – An ongoing challenge of interdisciplinary work





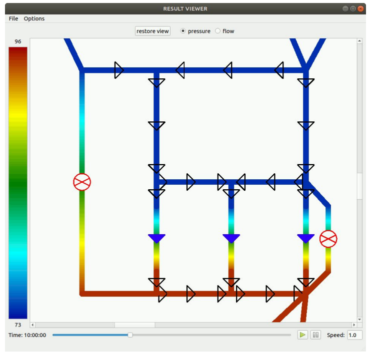
The CRC154: All PIs/PostDocs/PhDs

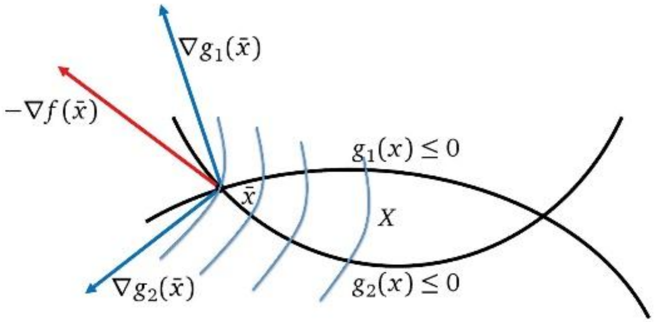


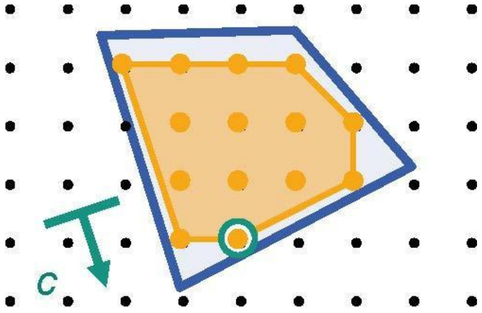


Visit trr154.fau.de

Future Work – An ongoing challenge of interdisciplinary work







Modeling and numerical simulation

- existence, uniqueness, regularity
- efficient algorithms, convergence, error control

Nonlinear optimization and control

- efficient algorithms, convergence, error control
- local optima and their characterization

Challenge: Coupling integer and continuous

- active research area
- general methods out-of-reach

Improvement continues …

Thanks to my co-authors

Robert Burlacu, Björn Geißler, Martin Gugat, Lukas Hümbs, Richard Krug, Jens Lang, Günter Leugering,

Antonio Morsi, Lars Schewe, Mathias Sirvent, Martin Schmidt, Dieter Weninger, David Wintergerst

Thanks to you for coming

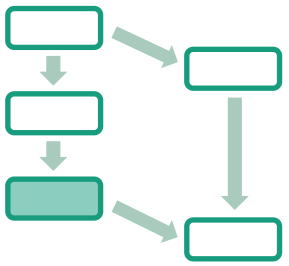
Alexander Martin

alexander.martin@utn.de

alexander.martin@iis.fraunhofer.de

TRR 154 · UTN · FAU Friedrich-Alexander-Universität Erlangen-Nürnberg · Fraunhofer IIS

Time Domain Decomposition – Hyperbolic semilinear equations



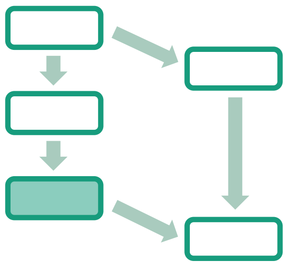
Time Domain Decomposition

We consider the two-point boundary value problem for a system of hyperbolic semilinear equations

$$\begin{aligned}
\min_{u,v,y} \quad & J(u,v,y) \\
\text{s.t.} \quad & \partial_t y + A(t,x)\partial_x y = f(t,x,u,y), \\
& y^+(t,0) = \sum_i g_i^0\left(t, y_i^-(t,0)\right), \\
& y^-(t,1) = \sum_j g_j^1\left(t, v(t), y_j^+(t,1)\right), \\
& y(0,x) = y_0(x), \quad u(t) \in U_{\mathrm{ad}}^{\mathrm{d}}, \quad v(t) \in U_{\mathrm{ad}}^{\mathrm{b}}
\end{aligned}$$

with the state y, distributed control u boundary control v $t \in [0, T]$nd $x \in [0,1]$

Time Domain Decomposition – Split the problem



Time Domain Decomposition

We decompose the time domain $[0, T]$ into finitely many non-overlapping sub-domains with

$$0 = t_0 < t_1 < \cdots < t_k < t_{k+1} < \cdots < t_K < t_{K+1} = T.$$

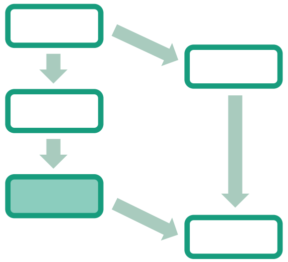
For each sub-domain $[t_k, t_{k+1}]$ we can state optimality conditions and iteratively decouple them via

$$\begin{aligned} y_k^{n+1}(t_{k+1}) + \beta p_k^{n+1}(t_{k+1}) &= \phi_{k,k+1}^n, \\ y_k^{n+1}(t_k) - \beta p_k^{n+1}(t_k) &= \phi_{k,k-1}^n \end{aligned}$$

and the update rule

$$\begin{aligned} \phi_{k,k+1}^n &= (1-\varepsilon)\left(y_{k+1}^n(t_{k+1}) + \beta p_{k+1}^n(t_{k+1})\right) \\ &\quad + \varepsilon\left(y_k^n(t_{k+1}) + \beta p_k^n(t_{k+1})\right), \\ \phi_{k,k-1}^n &= (1-\varepsilon)\left(y_{k-1}^n(t_k) - \beta p_{k-1}^n(t_k)\right) \\ &\quad + \varepsilon\left(y_k^n(t_k) - \beta p_k^n(t_k)\right). \end{aligned}$$

Time Domain Decomposition – Virtual Control Problems



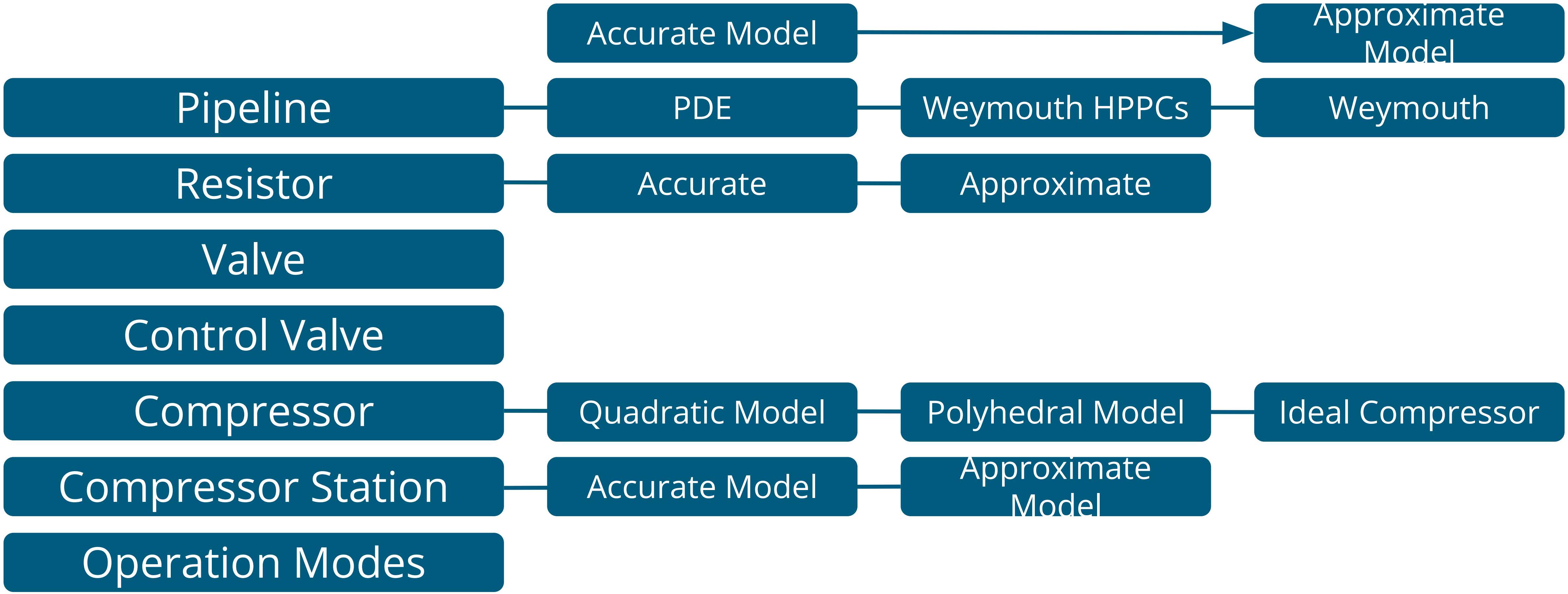
Time Domain Decomposition

The distinctive feature of this approach is that all sub-domain problems can be solved in parallel and have a primal interpretation as so-called *virtual control problems*

$$
\begin{aligned}
\min_{g_{k,k-1},u_k,v_k,y_k} \quad & J_k(u_k, v_k, y_k) + \frac{1}{2\beta}\left(\|y_k(t_{k+1}) - \phi_{k,k+1}\|_{L^2}^2 + \|g_{k,k-1}\|_{L^2}^2\right) \\
s.t. \quad & \partial_t y_k + A(t,x)\partial_x y_k = f_k(t, x, u_k, y_k), \\
& y_k^+(t,0) = \sum_i g_i^0(t, y_{ki}^-(t,0)), \\
& y_k^-(t,1) = \sum_j g_j^1(t, v_k(t), y_{kj}^+(t,1)), \\
& y_k(t_k, x) = \phi_{k,k-1} + g_{k,k-1}, \quad u_k(t) \in U_{\mathrm{ad}}^{\mathrm{d}}, \quad v_k(t) \in U_{\mathrm{ad}}^{\mathrm{b}}
\end{aligned}
$$

with virtual control $g_{k,k-1}$

Typical approach: Hierarchical modeling and solving



Pipeline: Euler differential equation

- **Pipeline** — PDE — Weymouth HPPCs — Weymouth
- Resistor
- Valve
- Control Valve
- Compressor
- Compressor Station
- Operation Modes

Euler-differential equation & equation of state for real gases:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho v)}{\partial x} = 0$$

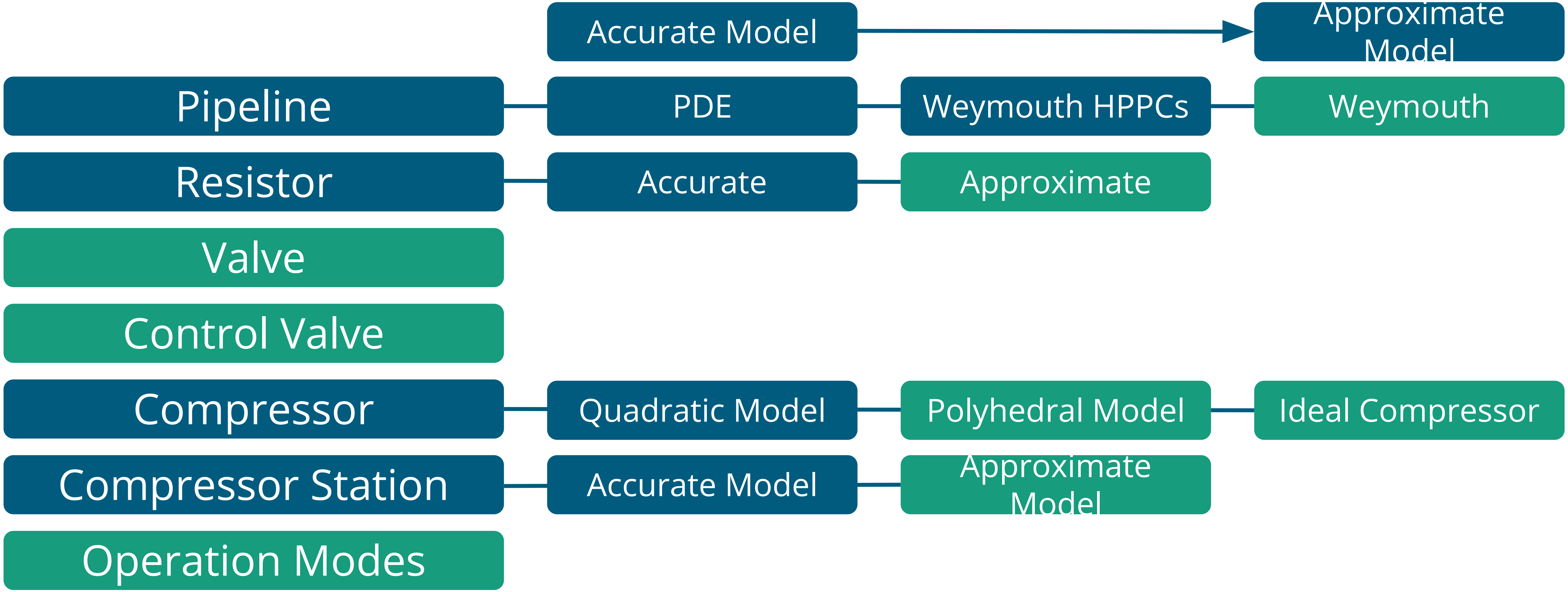
$$\frac{\partial (\rho v)}{\partial t} + \frac{\partial (\rho v^2)}{\partial x} + \frac{\partial p}{\partial x} + g\rho \frac{\partial h}{\partial x} + \lambda(q)\frac{|v|v}{2D}\rho = 0$$

$$A\rho c_p \left(\frac{\partial T}{\partial t} + v\frac{\partial T}{\partial x}\right) - A\left(1 + \frac{T}{z}\frac{\partial z}{\partial T}\right)\frac{\partial p}{\partial t}$$

$$-Av\frac{T}{z}\frac{\partial z}{\partial T}\frac{\partial p}{\partial x} + Avg\rho\frac{dh}{dx} + Q_E = 0$$

$$\rho - \frac{\rho_0 z_0 T_0}{p_0} \cdot \frac{p}{z(p,T)T} = 0$$

On the coarsest level: We end up with a MINLP



The CRC/TRR 154

- **Title**
  „Mathematical Modeling, Simulation, and Optimization using the Example of Gas Networks“

- **Partner**
  - FAU (spokes university)
  - HU Berlin, TU Berlin, WIAS, ZIB, TU Darmstadt, Uni Duisburg-Essen

- **Goal**
  A wholistic understanding of the Input/Output behavior of globally controlled dynamic networks