Mixed-Integer PDE-Constrained Optimization

GIAN Short Course on Optimization:
Applications, Algorithms, and Computation

Pelin Cay, Bart van Bloemen Waanders, Drew Kouri,
Anna Thuenen and Sven Leyffer

Lehigh University, Universität Magdeburg, Argonne National Laboratory,
and Sandia National Laboratories

September 12-24, 2016

Outline

Mixed-Integer PDE-Constrained Optimization (MIPDECO)

PDE-constrained MIP ... $u = u(t, x, y, z) \Rightarrow$ infinite-dimensional!

- t is time index; x, y, z are spatial dimensions

$$
\begin{cases}
\underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\
\text{subject to} & \mathcal{C}(u, w) = 0 \\
 & u \in \mathcal{U}, \quad \text{and} \quad w \in \mathbb{Z}^p \text{ (integers)},
\end{cases}
$$

- $u(t, x, y, z)$: PDE states, controls, & design parameters
- w discrete or integral variables

MIPDECO Warning

$w = w(t, x, y, z) \in \mathbb{Z}$ may be infinite-dimensional integers!

Mixed-Integer PDE-Constrained Optimization (MIPDECO)

PDE-constrained MIP ... $u = u(t, x, y, z) \Rightarrow$ infinite-dimensional!

- t is time index; x, y, z are spatial dimensions

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0 \\ & u \in \mathcal{U}, \quad \text{and} \quad w \in \mathbb{Z}^p \text{ (integers)}, \end{cases}$$

- $u(t, x, y, z)$: PDE states, controls, & design parameters
- w discrete or integral variables

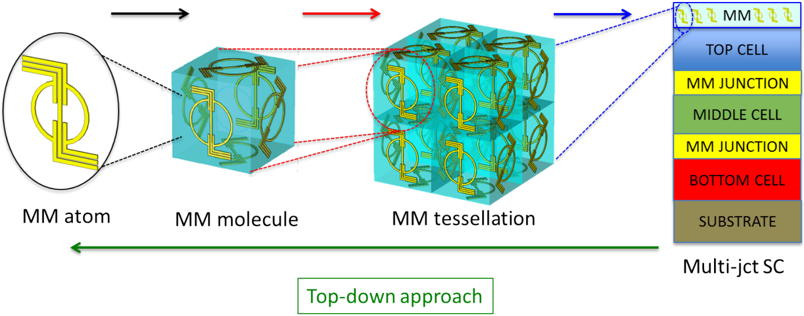
MIPDECO Warning

$w = w(t, x, y, z) \in \mathbb{Z}$ may be infinite-dimensional integers!

The Darth Vader of Optimization!

Design of Ultra-Efficient Solar Cell

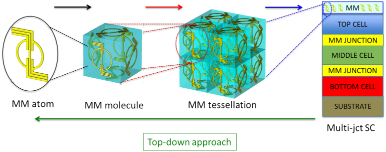
Design of non-reciprocal optical metamaterial for solar cells



Choose orientation of atoms and molecules to maximize energy

Design of Ultra-Efficient Solar Cell

Design of non-reciprocal optical metamaterial for solar cells



$$\nabla \times \mathbf{H} = -i\omega(\chi\mathbf{H} + \epsilon\mathbf{E}) + \mathbf{J}_e,$$
$$\nabla \times \mathbf{E} = i\omega(\mu\mathbf{H} + \zeta\mathbf{E}) + \mathbf{J}_m,$$

- Maxwell's equation gives **E** and **H** electric and magnetic field
- Objective is to maximize power inside solar cell (x space dims)

$$\frac{1}{2}\int_\omega I_{\text{solar}}(\omega)\int_V \Im(\epsilon(x,w))|\mathbf{E}(x,w;\omega)|^2 + \Im(\mu(x,w))|\mathbf{H}(x,w;\omega)|^2\, dV\, d\omega$$

- $w_{i,j,k} = 1$ if orientation i chosen on face j of molecule k
- $w_{i,j,k}$ impact permittivities and permeabilities in Maxwell's

$$\widetilde{\epsilon_{j,k}} = \sum_{i\in\mathcal{O}} w_{i,j,k}\epsilon_i$$

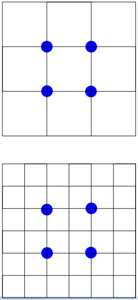
Mesh-Independent & Mesh-Dependent Integers

Definition (Mesh-Independent & Mesh-Dependent Integers)

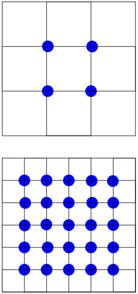
1. The integer variables are mesh-independent, iff number of integer variables is independent on the mesh.
2. The integer variables are mesh-dependent, iff the number of integer variables depends on the mesh.

Mesh-Independent

Manageable tree
Theory possible



Mesh-Dependent



Exploding tree
Theory???

Theoretical Challenges of MIPDECO

Functional Analysis (Mesh-Dependent w)

Denis Ridzal (Yoda): Function space which $w(x, y) \in \{0, 1\}$ in lies, you think?

- Consistently approximate $w(x, y) \in \{0, 1\}$ as $h \to 0$?
- Conjecture: $\{w(x, y) \in \{0, 1\}\} \neq L_2(\Omega)$
  ... e.g. binary support of Cantor set not integrable
- Likely need additional regularity assumptions

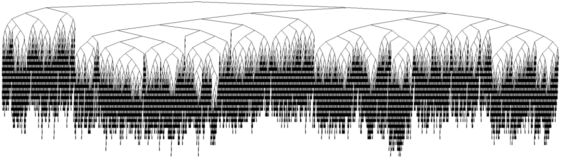
Coupling between Discretization & Integers

Discretization scheme (e.g. upwinding for wave equation) depends on direction of flow (integers).

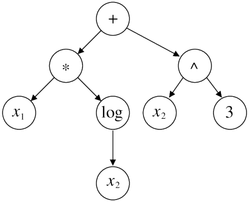
- Application: gas network models with flow reversals

Computational Challenges of MIPDECO

- Approaches for humongous branch-and-bound trees
  ... e.g. 3D topology optimization with 10^9 binary variables



- Warm-starts for PDE-constrained optimization (nodes)
  ... iterative Krylov (PDE) solve vs. rank-one updates (MIP)
- Guarantees for nonconvex (nonlinear) PDE constraints
  ... factorable programming approach hopeless for 10^9 vars!



... $f(x_1, x_2) = x_1 \log(x_2) + x_2^3$

Outline

Using My Favorite Lightsaber (Hammer)

Infinite-Dimensional MIPDECO

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u,w) \\ \text{subject to} & \mathcal{C}(u,w) = 0 \\ & u \in \mathcal{U}, \quad \text{and} \quad w \in \mathbb{Z}^p \quad \text{integer}, \end{cases}$$

Discretize $\Rightarrow$ finite dim. MINLP:

$$\begin{cases} \underset{x}{\text{minimize}} & f(x) \\ \text{subject to} & c(x) \leq 0 \\ & l \leq x \leq u, \quad x_i \in \mathbb{Z} \text{ for all } i \in I \end{cases}$$

Feel the force of AMPL/MINLP!

With abuse of notation ...

- x discretized (u, w)
- $f(x)$ discretized $\mathcal{F}(u, w)$
- $c(x)$discretized $\mathcal{C}(u, w)$

Source Inversion as MIP with PDE Constraints

Simple Example: Locate number of sources to match observation $\bar{u}$

$$\begin{cases} \underset{u,w}{\text{minimize}} \quad \mathcal{J} = \frac{1}{2}\int_\Omega (u-\bar{u})^2 d\Omega & \text{least-squares fit} \\ \text{subject to } -\Delta u = \sum_{k,l} w_{kl} f_{kl} \quad \text{in} \quad \Omega & \text{Poisson equation} \\ \qquad \sum_{k,l} w_{kl} \leq S \text{ and } w_{kl} \in \{0,1\} & \text{source budget} \end{cases}$$

with Dirichlet boundary conditions $u = 0$ on $\partial\Omega$.

E.g. Gaussian source term, $\sigma > 0$, centered at (x_k, y_l)

$$f_{kl}(x,y) := \exp\left(\frac{-\|(x_k, y_l) - (x,y)\|^2}{\sigma^2}\right),$$

Motivated by porous-media flow application to determine number of boreholes, [Ozdogan, 2004, Fipki and Celi, 2008]

Discretizing Poisson Equation

PDE (Poisson equation) in two dimensions (x, y)

$$-\Delta u = f \quad \Leftrightarrow \quad -\frac{\partial^2 u}{\partial x^2}(x,y) - \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y)$$

Discretize PDE on finite mesh, e.g. for $(x, y) \in \Omega = [0, 1]^2$

- Discretize $[0, 1]^2$ using mesh-size $h > 0$, e.g. $h = \frac{1}{N}$
- Meshpoints: $(x_i, y_j) = (ih, jh)$ for $i = 0, \ldots, N$, $j = 0, \ldots, N$
- Approximation of solution $u_{i,j} \approx u(x_i, y_j) = u(ih, jh)$

Second derivative central difference approximation

$$\frac{\partial^2 u}{\partial x^2}(x_i, y_j) \approx \frac{u_{i-1,j} - 2u_{i,j} + u_{i+1,j}}{h^2}$$

Similar for $u_{yy} \approx \left(u_{i,j-1} - 2u_{i,j} + u_{i,j+1}\right)/h^2$ ...

Source Inversion as MIP with PDE Constraints

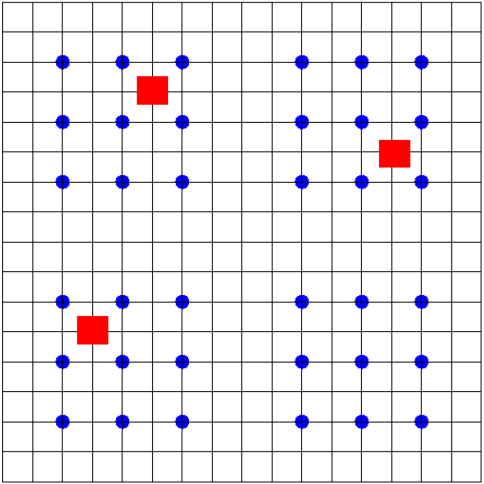
Consider 2D example with $\Omega = [0,1]^2$ and discretize PDE:

- 5-point finite-difference stencil; uniform mesh $h = 1/N$
- Denote $u_{i,j} \approx u(ih, jh)$ approximation at grid points

$$\begin{cases} \underset{u,w}{\text{minimize}} \quad J_h = \dfrac{h^2}{2} \displaystyle\sum_{i,j=0}^{N} (u_{i,j} - \bar{u}_{i,j})^2 \\ \text{subject to } \dfrac{4u_{i,j} - u_{i,j-1} - u_{i,j+1} - u_{i-1,j} - u_{i+1,j}}{h^2} = \displaystyle\sum_{k,l=1}^{N} w_{kl} f_{kl}(ih, jh) \\ \qquad u_{0,j} = u_{N,j} = u_{i,0} = u_{i,N} = 0 \\ \qquad \displaystyle\sum_{k,l=1}^{N} w_{kl} \leq S \text{ and } w_{kl} \in \{0,1\} \end{cases}$$

... finite-dimensional (convex) MIQP!

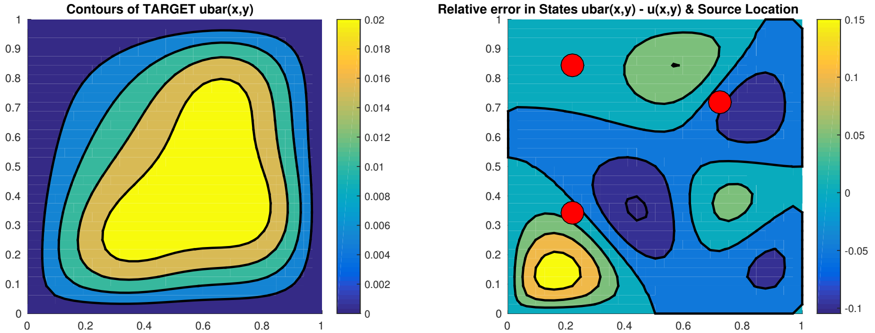
Mesh-Independent Source Inversion



Potential source locations (blue dots) on 16×16 mesh

Create target $\bar{u}$ using red square sources

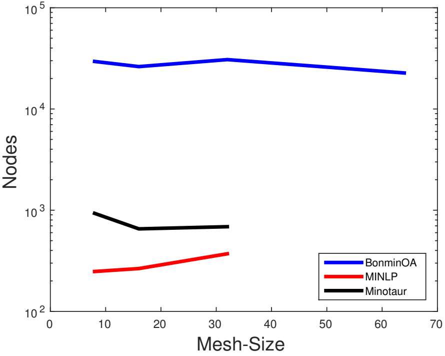
Source Inversion as MIP with PDE Constraints

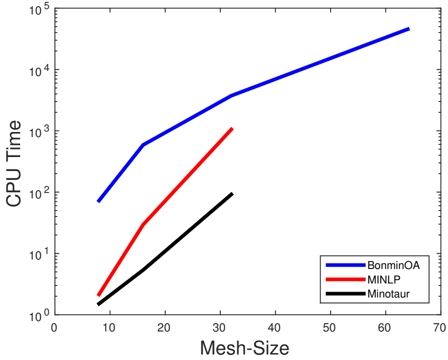


Target (3 sources), reconstructed sources, & error on 32×32 mesh

Mesh-Independent Source Inversion: MINLP Solvers

Number of Nodes and CPU time for Increasing Mesh Sizes

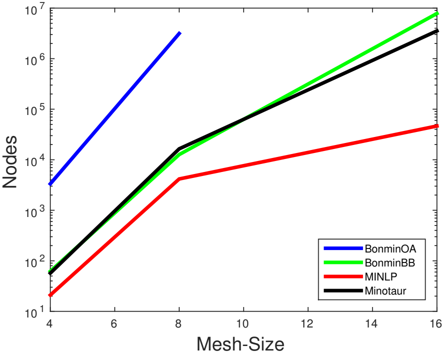


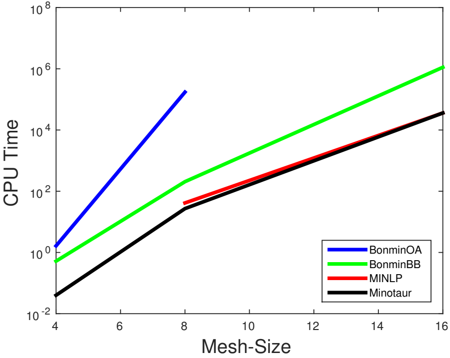


- Number of Nodes independent of mesh size!
- MINLP & Minotaur: filterSQP runs out of memory for $N \geq 32$
- BonminOA takes roughly 100 iterations ... quadratic objective

Mesh-Dependent Source Inversion: MINLP Solvers

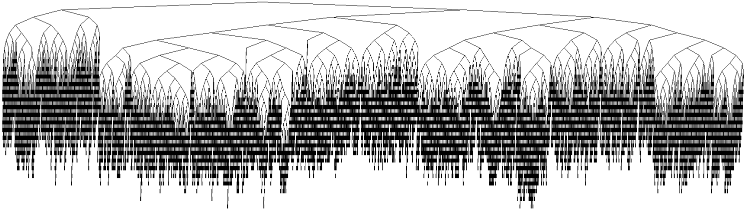
Number of Nodes and CPU time for Increasing Mesh Sizes





- Number of nodes & CPU time explodes with mesh size!
- OA `<BREAK>` after 130,000 seconds ... stress test for solvers!

MIPDECO trees are huge ... an overwhelming dark force!



MIPDECO Trick # 1: Eliminating the PDE

Discretized PDE constraint (Poisson equation)

$$\frac{4u_{i,j} - u_{i,j-1} - u_{i,j+1} - u_{i-1,j} - u_{i+1,j}}{h^2} = \sum_{k,l} w_{kl} f_{kl}(ih, jh), \ \forall i, j$$

$\Leftrightarrow A\mathbf{u} = \sum w_{kl}\mathbf{f}_{kl}$, where $w_{kl} \in \{0, 1\}$ only appear on RHS!

Elimination of PDE and states $u(x, y, z)$

- $A\mathbf{u} = \sum_{k,l} w_{kl}\mathbf{f}_{kl} \ \Leftrightarrow \ \mathbf{u} = A^{-1}\left(\sum_{k,l} w_{kl}\mathbf{f}_{kl}\right) = \sum_{k,l} w_{kl} A^{-1}\mathbf{f}_{kl}$
- Solve $n^2 \ll 2^n$ PDEs: $\mathbf{u}^{(kl)} := A^{-1}\mathbf{f}_{kl}$
- Eliminate $\mathbf{u} = \sum_{k,l} w_{kl}\mathbf{u}^{(kl)}$

MIPDECO Trick # 1: Eliminating the PDE

Eliminate $\mathbf{u} = \sum_{k,l} w_{kl} \mathbf{u}^{(kl)}$ in MINLP:

$$\begin{cases} \underset{w}{\text{minimize}} \quad J_h = \frac{h^2}{2} \sum_{i,j=0}^{N} \left(\sum_{k,l} w_{kl} \mathbf{u}_{ij}^{(kl)} - \bar{u}_{i,j} \right)^2 \\ \text{subject to} \sum_{k,l=1}^{N} w_{kl} \leq S \text{ and } w_{kl} \in \{0,1\} \end{cases}$$

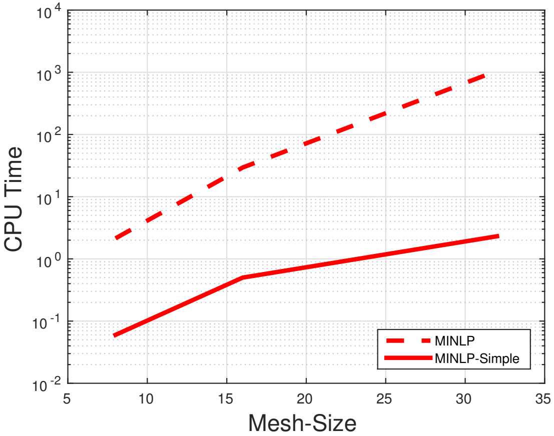
- Eliminates the states $\mathbf{u}$ (N^2 variables)
- Eliminates the PDE constraint (N^2 constraints)

... generalizes to other PDEs (with integer controls on RHS)

Simplified model is quadratic knapsack problem

Elimination of States & PDEs: Source Inversion

CPU Time for Increasing Mesh Sizes: Simplified vs. Original Model



Eliminating PDEs is two orders of magnitude faster!

Control Regularization: Not All Norms Are Equal

Poisson with Distributed Control [OPTPDE, 2014] & [Tröltzsch, 1984]

$$\begin{cases} \underset{u,w}{\text{minimize}} & \|u - u_d\|^2_{L^2(\Omega)} + \int_\Gamma e_\Gamma \; u \; ds + \alpha \, \|w\|^2_{L^x} \\ \text{subject to} & -\Delta u + u = w + e_\Omega \quad \text{in } \Omega \\ & \dfrac{\partial u}{\partial n} = 0 \text{ on boundary } \Gamma \\ & w(x,y) \in \{0,1\} \end{cases}$$

L^1 or L^2 regularization term for control $w(x,y) \in \{0,1\}$?

Good Norms for MIPs

MIP'ers prefer polyhedral norms ... promote integrality

- Old MIP trick: $w^2 = |w|$ for $w \in \{0,1\}$
  $\Rightarrow$ L^1-norm same as L^2-norm on binary variables!

Not All Norms Are Equal

Consider Distributed Control for increasing mesh-size

	CPU for L^2 Regularization			
Mesh	Minotaur	B-BB	B-Hyb	B-OA
8x8	0.04	0.80	2.54	126.81
16x16	6.61	72.21	1305.00	Time
32x32	Time	Time	Time	Time

Not All Norms Are Equal

Consider Distributed Control for increasing mesh-size

	CPU for L^2 Regularization			
Mesh	Minotaur	B-BB	B-Hyb	B-OA
8x8	0.04	0.80	2.54	126.81
16x16	6.61	72.21	1305.00	Time
32x32	Time	Time	Time	Time

	CPU for L^1 Regularization			
Mesh	Minotaur	B-BB	B-Hyb	B-OA
8x8	0.03	0.48	0.21	0.04
16x16	0.11	3.62	0.66	0.20
32x32	0.18	62.66	3.53	0.74

- L^1 regularization is equivalent to L^2, but faster
- Many fewer nodes in tree-searches $\Rightarrow$ solve up to 256×256

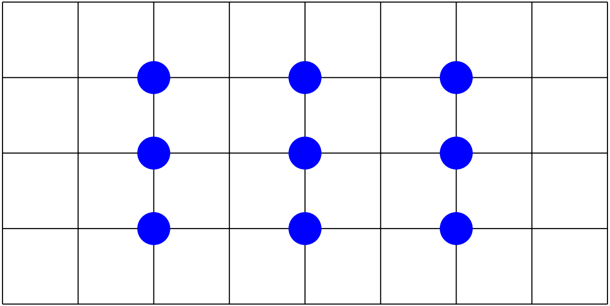
Outline

Actuator Placement and Operation [Falk Hante]

Goal: Control temperature with actuators

- Select sequence of control inputs (actuators)
- Choose continuous control (heat/cool) at locations
- Match prescribed temperature profile

... "de-mist bathroom mirror with hair-drier"



Potential Actuator Locations $l = 1, \ldots, L$

Actuator Placement and Operation

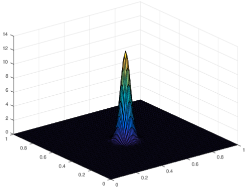
Find optimal sequence of actuators, $w_l(t)$, and controls, $v_l(t)$:

$$
\begin{cases}
\underset{u,v,w}{\text{minimize}} & \|u(t_f,\cdot)\|^2_\Omega + 2\|u\|^2_{T\times\Omega} + \frac{1}{500}\|v\|^2_T \\
\text{subject to} & \dfrac{\partial u}{\partial t} - \kappa\Delta u = \displaystyle\sum_{l=1}^{L} v_l(t) f_l \ \text{ in } T\times\Omega \\
 & w_l(t) \in \{0,1\}, \quad \displaystyle\sum_{l=1}^{L} w_l(t) \le W, \ \forall t \in T \\
 & L w_l(t) \le v_l(t) \le U w_l(t), \ \forall l = 1,\ldots,L, \ \forall t \in T
\end{cases}
$$

where

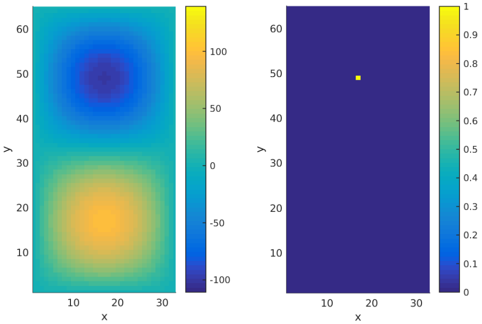
$$
f_l(x,y) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(\frac{-\|(x,y)-(x_l,y_l)\|^2)}{2\sigma} \right)
$$

point-source for actuators at (x_l, y_l)



Actuator Placement and Operation

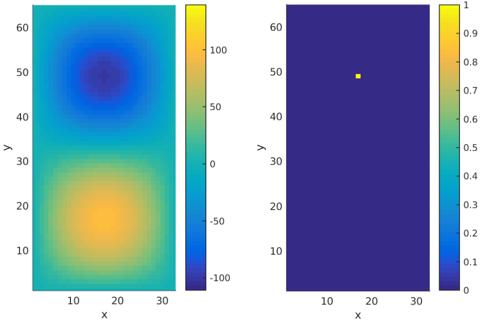
Solution of NLP relaxation on 32×32 mesh ...



... provides no useful information?

Actuator Placement and Operation

Solution of MIPDECO on 32×32 mesh ...



... implementable discrete control!

NLP Relaxations of MIPDECOs Can Be Useless

Relaxations useless

- Controls smeared out $w_t = \frac{1}{W}$
- No obvious rounding ...
  ... in fact, $\text{round}(w_t) = 0$
- Branch-and-bound is hopeless ...
  ... generate huge search tree
- Many identical subtrees
  $\Rightarrow$ exploit symmetry
  (orbital branching)

... can make heuristics work ...

Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...

Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...

Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

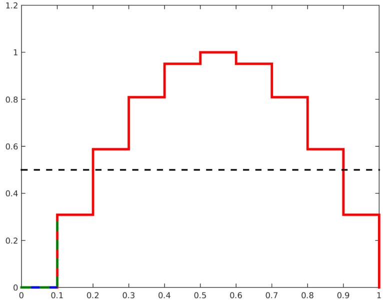
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...
Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

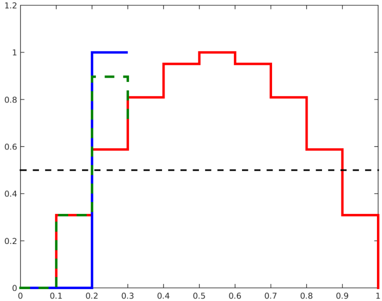
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...
Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

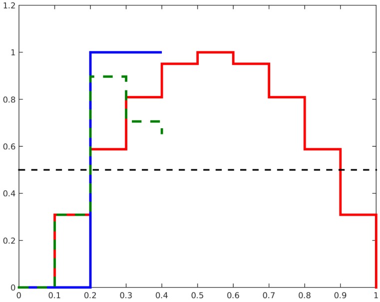
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} \left(\tilde{w}_\tau - w_t \right)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \ w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...

Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

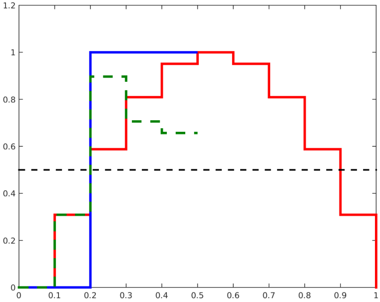
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} \left(\tilde{w}_\tau - w_t \right)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...
Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

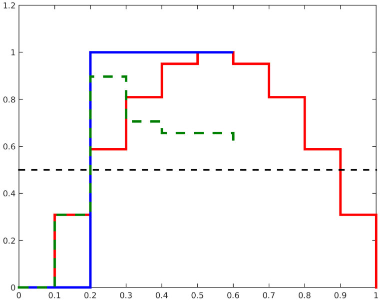
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \ w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...
Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

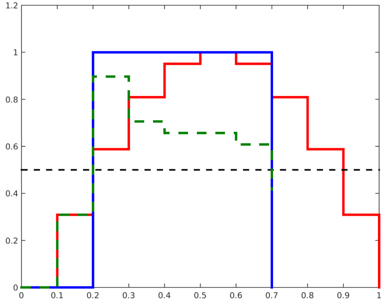
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} \left(\tilde{w}_\tau - w_t \right)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \ w(t) \in \{0,1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...
Let $\tilde{w}_t \in [0,1]$ continuous relaxation ... construct integral w_t

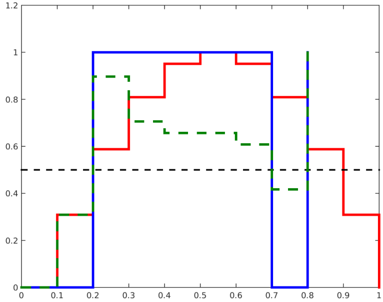
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...

Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

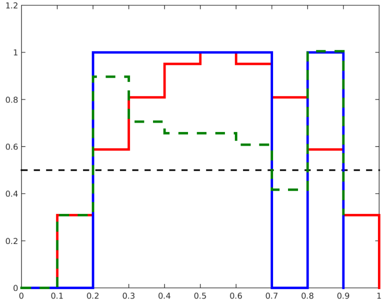
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

end



Sum-Up Rounding Heuristics for MIPDECOs

MIPDECO with binary control $w(t)$ independent of (x, y, z) ...

$$\begin{cases} \underset{u,w}{\text{minimize}} & \mathcal{F}(u, w) \\ \text{subject to} & \mathcal{C}(u, w) = 0, \quad u \in \mathcal{U}, \; w(t) \in \{0, 1\}^p \end{cases}$$

Generalize optimal-control sum-up-rounding [Sager et al., 2012] ...

Let $\tilde{w}_t \in [0, 1]$ continuous relaxation ... construct integral w_t

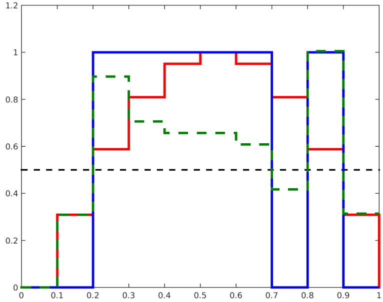
for $t = 1, \ldots, T$ **do**

Compute rounding residual:

$$r_t := \tilde{w}_t + \sum_{\tau=0}^{t-1} (\tilde{w}_\tau - w_t)$$

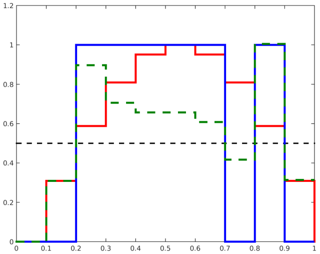
Round: $w_t = \begin{cases} 1 \text{ if } & r_t > \frac{1}{2} \\ 0 \text{ else} \end{cases}$

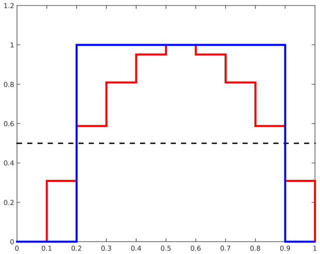
end



Sum-Up Rounding vs. Simple Rounding

Simple rounding: $w_t = \text{round}(\tilde{w}_t)$





Sum-Up Rounding: $6.31 \rightarrow 6$

Simple Rounding: $6.31 \rightarrow 7$

Simple Rounding arbitrarily poor: $\tilde{w}_t = 0.500001 \Rightarrow w_t = 1$

Sum-Up Rounding has guarantees on quality of bounds!

Results for Sum-Up Rounding

Consider 2D heat equation with Robin boundary control

- $\Omega = [0,1]^2 \times [0,2]$ discretized with $N = 8, 16, 32$ in space and $M = 16, 32, 64$ in time.
- MINLP solvers: Minotaur and Bonmin (BnB, Hyb, OA)
- Sum-Up-Rounding (knapsack): Two NLPs solved with IPOPT
- Two instances per mesh (different initial cond^s & forcing)

	Problem Size		
Mesh	# Variables	# Binary Vars	# Constraints
8x8x16	2873	272	3094
16x16x32	13497	528	13926
32x32x64	82745	1040	83590

Results for Sum-Up Rounding

Consider 2D heat equation with Robin boundary control

- $\Omega = [0,1]^2 \times [0,2]$ discretized with $N = 8, 16, 32$ in space and $M = 16, 32, 64$ in time.
- MINLP solvers: Minotaur and Bonmin (BnB, Hyb, OA)
- Sum-Up-Rounding (knapsack): Two NLPs solved with IPOPT
- Two instances per mesh (different initial cond^s & forcing)

	CPU Time [s] for Solution				
Mesh	Minotaur	B-BnB	B-Hyb	B-OA	SUR-k
8x8x16	4660.4	4660.4	4660.4	Time	1.08
8x8x16	18240.4	18240.4	18240.4	18240.4	1.66
16x16x32	Time	Time	Time	Time	23.7
16x16x32	4333.4	Time	4332.6	Time	43.5
32x32x64	Time	Time	Time	Time	9297.5
32x32x64	Time	Time	Time	Time	2650.7

Results for Sum-Up Rounding

NLPs solve faster than MINLPs ... what about solution quality?

Results for Sum-Up Rounding

NLPs solve faster than MINLPs ... what about solution quality?

	Solution Bounds and Gap		
N	Low Bnd	Upp Bnd	Rel. Gap
8	4660.4	4809.9	3.1%
8	18240.4	18838.6	3.2%
16	2483.6	2517.9	1.4%
16	4332.7	4840.1	10.5%
32	900.8	976.8	7.8%
32	1840.8	2560.5	28.1%

... most solutions within 10% of optimum!

Conclusions

Mixed-Integer PDE-Constrained Optimization (MIPDECO)

- Class of challenging problems with important applications
  - Subsurface flow: oil recovery or environmental remediation
  - Design of next-generation solar cells
- Classification: mesh-dependent vs. mesh-independent
- On-going work: Building AMPL library of test problems
  ... formulation matters: interplay of binary and continuous
- Discretized PDEs $\Rightarrow$ huge MINLPs ... push solvers to limit
- Elimination of PDE and state variables $u(t, x, y, z)$
- Sum-up rounding heuristics can be generalized

Outlook and Extensions

- Consider multi-level in space (network) and time
- Move toward truly multi-level approach similar to PDEs

The Deathstar of Optimization Problems ...

Add nonlinearities, uncertainty, robustness ...

Fipki, S. and Celi, A. (2008).
The use of multilateral well designs for improved recovery in heavy oil reservoirs.
In *IADV/SPE Conference and Exhibition*, Orlanda, Florida. SPE.

OPTPDE (2014).
OPTPDE — a collection of problems in PDE-constrained optimization.
http://www.optpde.net.

Ozdogan, U. (2004).
Optimization of well placement under time-dependent uncertainty.
Master's thesis, Stanford University.

Sager, S., Bock, H., and Diehl, M. (2012).
The integer approximation error in mixed-integer optimal control.
Mathematical Programming A, 133(1–2):1–23.

Tröltzsch, F. (1984).
The generalized bang-bang-principle and the numerical solution of a parabolic boundary-control problem with constraints on the control and the state.
Zeitschrift für Angewandte Mathematik und Mechanik, 64(12):551–556.