MPEC Strategies for Optimization of Chemical Process Dynamics

B. T. Baumrucker and L. T. Biegler
Chemical Engineering Department
Carnegie Mellon University
Pittsburgh, PA 15213

May, 2008

1

- Introduction
  - What are MPECs?
  - Challenges for optimization
- MPEC Properties and Algorithms
- 'Safe' MPEC Formulations
  - Guidelines
  - Examples
- MPECs for Hybrid Systems
  - Problem Class
  - Dynamic Optimization Examples
- Case Studies
  - Dynamic pipelines
  - Fermentation with Metabolic Networks

2

MPECs in Hybrid Systems Optimization

Goal: Incorporate discrete decisions within optimization problems

Generalized Disjunctive Programming & Mixed Integer Programming

- Most general method to handle logical disjunctions
- Computational expense may be high for large systems with many discrete decisions, such as dynamic systems with switches at any point in time

MPEC - Alternative for a limited class of disjunctive problems

+ can be used to model many common phenomena, including disappearance of phases, flow reversal, and hybrid dynamics
+ Can be embedded within a standard Non-Linear Programming solver and obtain fast local solutions
+ Computational cost scales polynomially with problem size
- Introduces an inherent non-convexity and constraint dependency

3

What is an MPEC?

Mathematical Program with Equilibrium Constraints

$$\begin{aligned} \min \quad & f(x,y) \\ s.t. \quad & (x,y) \in Z \\ & (v-y)^T F(x,y) \geq 0 \\ & y \in C(x) \quad \forall v \in C(x) \end{aligned}$$

$$(F(x,y) = \nabla_y \theta(x,y))$$

$$\begin{aligned} \min_x \quad & f(x,y) \\ s.t. \quad & (x,y) \in Z \\ & y = \arg\min_{\hat{y}} \{\theta(x,\hat{y}) : \hat{y} \in C(x)\} \end{aligned}$$

Substitute KKT conditions for inner problem with complementarities

$$\begin{aligned} \min \quad & f(x,y,z) \\ s.t. \quad & h(x,y,z) = 0 \\ & g(x,y,z) \geq 0 \\ & 0 \leq x \perp y \geq 0 \end{aligned}$$

$$\Leftrightarrow \begin{cases} x_i = 0 \quad \vee \quad y_i = 0 \\ x \geq 0, y \geq 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} x_i y_i = 0 \\ x \geq 0, y \geq 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} x_i y_i \leq 0 \\ x \geq 0, y \geq 0 \end{cases}$$

$$\Leftrightarrow \begin{cases} x^T y \leq 0 \\ x \geq 0, y \geq 0 \end{cases}$$

4

Why are MPECs hard?

$$\begin{aligned} \min \quad & f(x) \\ s.t. \quad & h(x) = 0 \\ & g(x) \geq 0 \\ & 0 \leq G(x) \perp H(x) \geq 0 \end{aligned}$$

$$\nabla f(x^*) - \nabla g(x^*)\lambda - \nabla h(x^*)\mu - \nabla G(x^*)\tau - \nabla H(x^*)\nu + [H(x^*)\nabla G(x^*) + G(x^*)\nabla H(x^*)]\delta = 0$$

$$\nabla f(x^*) - \nabla g(x^*)\lambda - \nabla h(x^*)\mu - \nabla G(x^*)[\tau - H(x^*)\delta] - \nabla H(x^*)[\nu - G(x^*)\delta] = 0$$

- Violates constraint qualifications (i.e. LICQ, MFCQ)
- Constraint multipliers will be non-unique and unbounded
- Existence of multipliers not guaranteed

$$\begin{aligned} \min \quad & f(x,y) \\ s.t. \quad & 0 \leq x \perp y \geq 0 \end{aligned} \quad \Rightarrow \quad \begin{aligned} \min \quad & f(x,y) \\ s.t. \quad & x \geq 0 \\ & y \geq 0 \\ & xy \leq 0 \end{aligned}$$

$$\text{Assume } x = 0 \text{ at solution} : c(x,y) = \begin{bmatrix} x \\ xy \end{bmatrix} = 0 \text{ with } \nabla c = \begin{bmatrix} 1 & 0 \\ y & 0 \end{bmatrix}$$

$\Rightarrow$ linearly dependent, violates LICQ

Specialized solvers must be used <u>or</u> problem must be modified to use standard NLP solvers

B-Stationarity

A point is B-Stationary if it is feasible to the MPCC and *d=0* solves the following LPEC (Linear Program with Equilibrium Constraints):

$$\begin{aligned} \min_d \quad & \nabla f(w^*)^T d \\ s.t. \quad & g(w^*) + \nabla g(w^*)^T d \geq 0 \\ & h(w^*) + \nabla h(w^*)^T d = 0 \\ & 0 \leq x^* + d_x \perp y^* + d_y \geq 0 \end{aligned}$$

where

$w^* = (x^*, y^*, z^*)$

$I_X = \{i : x_i^* = 0\}$

$I_Y = \{i : y_i^* = 0\}$

Requires solution of 2^m linear programs, where m is the dimension of the biactive set, $I_X \cap I_Y$

Strong Stationarity

A point is strongly stationary if it is feasible to the MPCC and *d=0* solves the following relaxed LP problem:

$$
\begin{aligned}
\min_{d} \quad & \nabla f(w^*)^T d \\
s.t. \quad & g(w^*) + \nabla g(w^*)^T d \geq 0 \\
& h(w^*) + \nabla h(w^*)^T d = 0 \\
& d_{xi} = 0, && i \in I_X \setminus I_Y \\
& d_{yi} = 0, && i \in I_Y \setminus I_X \\
& d_{xi} \geq 0, && i \in I_X \cap I_Y \\
& d_{yi} \geq 0, && i \in I_X \cap I_Y
\end{aligned}
$$

- MPEC-LICQ for MPCC ≡ LICQ of relaxed LP
- If MPEC-LICQ holds, B-stationary point equivalent to Strong Stationary (Anitescu, Tseng, Wright, 2007)
- Single set of NLP multipliers exists for MPCC solution

➔ MPCC can be solved through an NLP reformulation

7

Complementarity Reformulations

Reg(ε): $\begin{array}{ll} \min & f(x,y) \\ s.t. & 0 \leq x \perp y \geq 0 \end{array} \Rightarrow \lim_{\varepsilon \to 0} \begin{pmatrix} \min & f(x,y) \\ s.t. & x_i y_i \leq \varepsilon \\ & x, y \geq 0 \end{pmatrix}$

RegComp(ε): $\begin{array}{ll} \min & f(x,y) \\ s.t. & 0 \leq x \perp y \geq 0 \end{array} \Rightarrow \lim_{\varepsilon \to 0} \begin{pmatrix} \min & f(x,y) \\ s.t. & x^T y \leq \varepsilon \\ & x, y \geq 0 \end{pmatrix}$

RegEq(ε): $\begin{array}{ll} \min & f(x,y) \\ s.t. & 0 \leq x \perp y \geq 0 \end{array} \Rightarrow \lim_{\varepsilon \to 0} \begin{pmatrix} \min & f(x,y) \\ s.t. & x_i y_i = \varepsilon \\ & x, y \geq 0 \end{pmatrix}$

PF(ρ): $\begin{array}{ll} \min & f(x,y) \\ s.t. & 0 \leq x \perp y \geq 0 \end{array} \Rightarrow \begin{array}{ll} \min & f(x,y) + \rho x^T y \\ s.t. & x, y \geq 0 \end{array}$

Note: Reg(ε), RegComp(ε), and RegEq(ε) require the solution of a series of problems. PF(ρ) can be used by any general NLP solver for fixed ρ sufficiently large.

Ralph, D. and S. J. Wright, *Optimization Methods and Software*, 19, 5, pp. 527-556 (2004)

8

MPECLib Results

92 Test Problems from academia and industry including both small and large problems

Library available in GAMS format – maintained by MPEC World part of GAMS World

Compare performance of different solvers with different MPEC reformulations

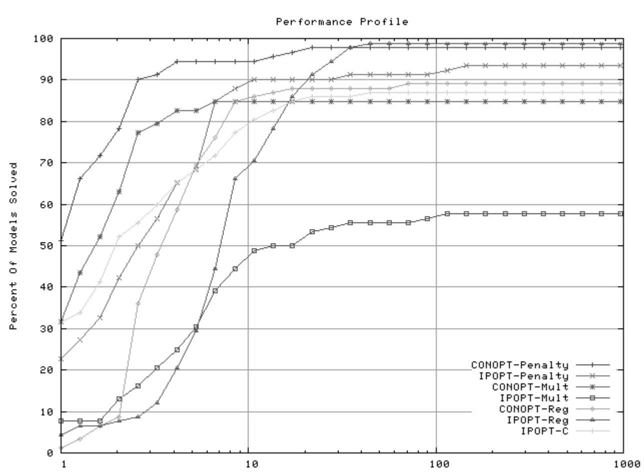
- Solvers including: CONOPT, IPOPT, IPOPT-C
- Reformulations including: Penalty, Mult, Reg

22 of 92 problems only one solution was found by all methods that converged

Results in Dolan-Morè Plot

9

92 Problem Test Set



10

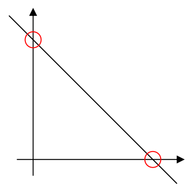
A Bad Formulation – EXOR Model

Suppose we want (x,y) to take the value (0,1) or (1,0)

- Consider the formulation:

$$x + y = 1$$
$$0 \le x \perp y \ge 0$$

- The feasible region graphically:



Disjoint Feasible Region!

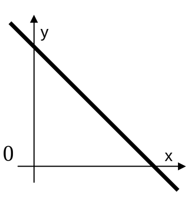
MPCC Formulation Guidelines

Convex inner minimization problem leading to complementarity.

Outer constraints should preserve connected feasible region for inner problem

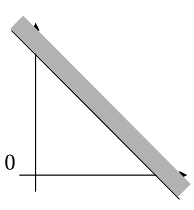
$$x + y = 1$$
$$\min_x \ \phi(y)x \quad s.t. \ 0 \le x \le 1$$

$$x + y = 1$$
$$\phi(y) = \lambda_+ - \lambda_-$$
$$0 \le x \perp \lambda_- \ge 0$$
$$0 \le (1-x) \perp \lambda_+ \ge 0$$



$$x + y \le 1$$
$$\min_x \ \phi(y)x \quad s.t. \ 0 \le x \le 1$$

$$x + y \le 1$$
$$\phi(y) = \lambda_+ - \lambda_-$$
$$0 \le x \perp \lambda_- \ge 0$$
$$0 \le (1-x) \perp \lambda_+ \ge 0$$



MPCC Formulation Guidelines

- Define inner minimization problem
- Verify that outer problem constraints lead to connected feasible regions for inner problem variables
- Convert MPEC to MPCC by writing out optimality conditions
- Simplify resulting expression
- Solve MPCC using NLP reformulation

Example: $y = |f(x)|$

$$\max \; f(x)v, \quad s.t. \; -1 \le v \le 1$$

$$f(x) = s_+ - s_-$$
$$-1 \le v \le 1$$
$$0 \le s_+ \perp 1 - v \ge 0$$
$$0 \le s_- \perp 1 + v \ge 0$$
$$y = f(x)v$$

$$v = -1, s_- = 0, s_+ \ge 0$$
$$v = 1, s_+ = 0, s_- \ge 0$$

$$0 \le s_- \perp s_+ \ge 0$$
$$y = s_+ + s_-$$
$$f(x) = s_+ - s_-$$

13

Generalized Complementarity Formulations

Abs(*)

$$\begin{matrix} f(x) = s^+ - s^- \\ 0 \le s^+ \perp s^- \ge 0 \end{matrix} \Rightarrow |f(x)| = s^+ + s^-$$

Min(*,*) & Max(*,*) (includes Pos(*), Neg(*))

$$\begin{matrix} y = \min(f(x), y^{UB}) \\ f(x) - y = s \\ y \le y^{UB} \perp s \ge 0 \end{matrix} \qquad \begin{matrix} y = \max(f(x), y^{LB}) \\ f(x) - y = s \\ y \ge y^{LB} \perp s \ge 0 \end{matrix} \qquad \begin{matrix} Pos(f(x)) = \max(f(x), 0) \\ Neg(f(x)) = \min(f(x), 0) \end{matrix}$$

Signum(*)

$$signum(x) = \begin{cases} -1 & x < 0 \\ 1 & x > 0 \end{cases} \Rightarrow \begin{matrix} \min & -u * x \\ s.t. & -1 \le u \le 1 \end{matrix} \Rightarrow signum(x) = u$$

If * Then * Else * (includes Piecewise Functions but not EXOR)

$$\begin{matrix} \min & u(x - x_{switch}) \\ s.t. & 0 \le u \le 1 \end{matrix}$$
$$y = (u) f_1(x) + (1-u) f_2(x)$$

$$(x - x_{switch}) - \lambda_0 + \lambda_1 = 0$$
$$0 \le \lambda_0 \perp u \ge 0$$
$$0 \le \lambda_1 \perp (1-u) \ge 0$$

14

Optimal Design for Distillation

Building on Previous Work

- Complementarity relaxation of equilibrium derived from minimization of Gibbs free energy (Gopal, B., 1999)
  - Applied to flash problems
  - Solved via smoothing methods with CONOPT
- Complementarity relaxation applied to trayed distillation (Raghunathan, B., 2005)
  - Steady state and dynamic columns operation optimized
  - Solved via Reg(t) and interior point method IPOPT-C
- Complementarity relaxation used in conjunction with Distributed Stream Method
  - Optimal design and operation of tray columns (Lang, B., 2004)
  - Solved via smoothing methods with CONOPT

MPEC Formulation for Distillation Optimization

(Baumrucker Renfro, B., 2008)

$$(L_i + DL + rd)x_{ij} + DVy_{ij} = V_{i-1}y_{i-1,j} \quad i \in CON$$

$$L_i x_{ij} + V_i y_{ij} = L_{i+1}x_{i+1,j} + V_{i-1}y_{i-1,j} + \sum_k f_{ik} Fd_k xf_{dj} + g_i \cdot rd \cdot x_{ij} \quad i \in COL$$

$$Bx_{ij} + V_i y_{ij} = L_{i+1}x_j + \sum_k f_{ik} Fd_k xf_{dj} \quad i \in REB$$

$$y_{ij} = \beta_i K_{ij} x_{ij}$$

$$\beta_i = 1 - s_i^l + s_i^v$$

$$0 \le L_i \perp s_i^l \ge 0$$

$$0 \le V_i \perp s_i^v \ge 0$$

$$(L_i + DL + rd)hl_i + DV \cdot hv_i = V_{i-1}hv_{i-1} + Q_c \quad i \in CON$$

$$L_i hl_i + V_i hv_i = L_{i+1}hl_{i+1} + V_{i-1}hv_{i-1} + \sum_k f_{ik} Fd_k shf_{dj} + g_i \cdot rd \cdot hl_i \quad i \in COL$$

$$B \cdot hl_i + V_i hv_i = L_{i+1}hl_{i+1} + \sum_k f_{ik} Fd_k shf_{dj} + Q_H \quad i \in REB$$

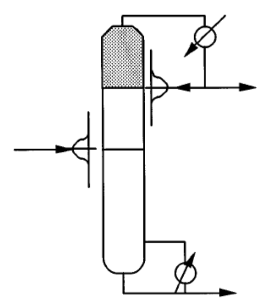
$$\sum_J x_{ij} - \sum_J y_{ij} = 1$$

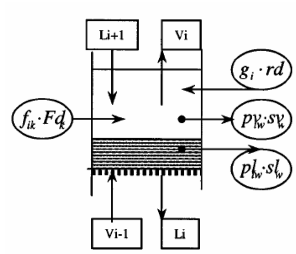
$$R_{total} = R \cdot D$$

$$L_{N_{\max}} = (1 - rdf)R_{total}$$

$$rd = rdf \cdot R_{total}$$

$$f_{ik} = \frac{\exp\left[-\left(\frac{i - Nf_k}{\sigma_k}\right)^2\right]}{\sum_j \exp\left[-\left(\frac{j - Nf_k}{\sigma_k}\right)^2\right]} \quad i \in COL \cup REB, k \in K \qquad g_{ik} = \frac{\exp\left[-\left(\frac{i - Nt}{\sigma_r}\right)^2\right]}{\sum_j \exp\left[-\left(\frac{j - Nt}{\sigma_r}\right)^2\right]} \quad i \in COL$$





Distillation Results

Solver determines optimal feed tray location and operating conditions

Objective functions (min):

$$obj = wt \cdot DL \cdot x('25','Toluene') + wr \cdot r + wn \cdot Nt - rdf$$

$$obj = wr \cdot r + wn \cdot Nt$$

Benzene-Toluene

	Smoothing	Penalty (ρ=1000)
Iterations	639	287
CPU seconds	25	9
Objective Function Value	3.2643	3.2643
Variables	359	361
Equations	352	304

17

MPECs for Hybrid Dynamics

$$\varsigma(x(t),u(t),t) > 0,\ \dot{x} = f_{+}(x,u,t)$$
$$\varsigma(x(t),u(t),t) < 0,\ \dot{x} = f_{-}(x,u,t)$$
$$\varsigma(x(t),u(t),t) = 0,\ \dot{x} = v(t) f_{-}(x,u) + (1 - v(t)) f_{+}(x,u), v(t) \in [0,1]$$

- Switching functions determine transition to different state models
- Transitions embedded within DAEs, differential states remain continuous
- Non-vanishing epochs of variable time, with (index 1) DAE models
- Transitions at epoch boundaries only

$$\bigvee_{m \in M} \left[\begin{array}{c} \left. \begin{array}{l} F^m(\dot{x}(t), x(t), y(t), u(t), v_i, p) = 0 \\ g^m(x(t), y(t), u(t), p) \le 0 \end{array} \right\} t \in [t_i, t_{i+1}] \\ Y_m \\ x(t_i^-) = x(t_i^+) \\ w^m(x(t), v_i) = 0, t \in [t_i, t_{i+1}] \end{array} \right]$$
$$H(Y) = True,\ i = 0, \ldots N$$

18

MPECs for Hybrid Systems

$$x(t_i^-) = x(t_i^+) \rightarrow \left. \begin{array}{l} F^m(\dot{x}(t), x(t), y(t), u(t), v_i, p) = 0 \\ g^m(x(t), y(t), u(t), p) \le 0 \end{array} \right\} t \in [t_i, t_{i+1}] \quad \underset{m \in M}{\vee} w^m(x(t), v_i) = 0,\ t \in [t_i, t_{i+1}] \rightarrow t_{i+1}$$

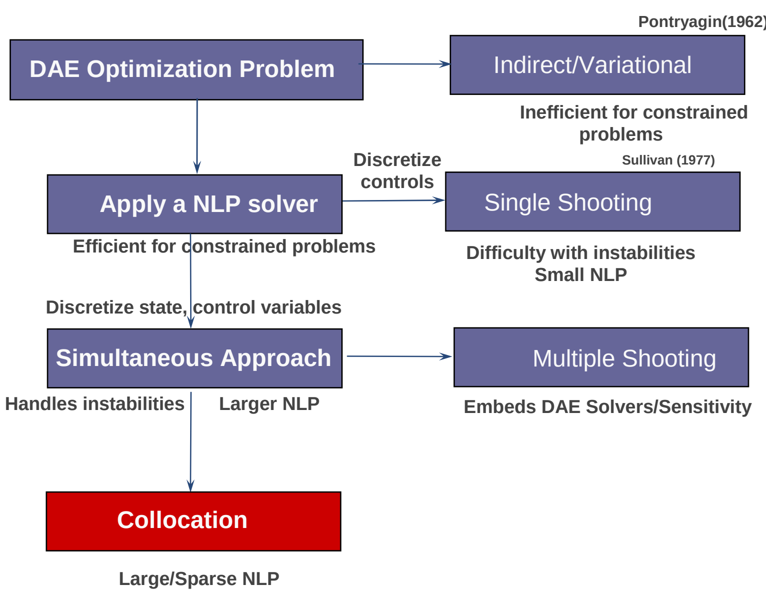
- Switching functions determine transition to different state models
- Transitions embedded within DAEs, differential states remain continuous
- Non-vanishing epochs of variable time, with (index 1) DAE models
- Transitions at epoch boundaries only

$$\underset{m \in M}{\vee} \begin{bmatrix} \left. \begin{array}{l} F^m(\dot{x}(t), x(t), y(t), u(t), v_i, p) = 0 \\ g^m(x(t), y(t), u(t), p) \le 0 \end{array} \right\} t \in [t_i, t_{i+1}] \\ Y_m \\ x(t_i^-) = x(t_i^+) \\ w^m(x(t), v_i) = 0, t \in [t_i, t_{i+1}] \end{bmatrix}$$

$$H(Y) = True,\ i = 0, \ldots N$$

19

Dynamic Optimization Approaches



20

Comparison of Computational Complexity

($\alpha \in [2, 3]$, $\beta \in [1, 2]$, n_w, n_u - assume $N_m = O(N)$)

	Single Shooting (SQP/NPSOL)	Multiple Shooting (rSQP/SNOPT)	Simultaneous Collocation (Barrier/IPOPT)
DAE Integration	$n_w^\beta N$	$n_w^\beta N$	---
Sensitivity	$(n_w N)(n_u N)$	$(n_w N)(n_u + n_w)$	$N(n_u + n_w)$
Exact Hessian	$(n_w N)(n_u N)^2$	$(n_w N)(n_u + n_w)^2$	$N(n_u + n_w)$
NLP Decomposition	---	$n_w^3 N$	---
Step Determination	$(n_u N)^\alpha$	$(n_u N)^\alpha$	$((n_u + n_w)N)^\beta$
Backsolve	---	---	$((n_u + n_w)N)$
	$O((n_uN)^\alpha + N^2 n_w n_u + N^3 n_w n_u^2)$	$O((n_uN)^\alpha + N n_w^3 + N n_w (n_w + n_u)^2)$	$O((n_u + n_w)N)^\beta$

21

Hybrid Optimization Problem

$$Min\ \Phi(x(t_{N+1}))$$

$$s.t.\ u(t) \in U, x(t) \in X, y(t) \in Y, v^j(t) \in [0,1], j = 1,\ldots n_v$$

$$\bigvee_{m \in M} \begin{bmatrix} \left.\begin{matrix} F^m(\dot{x}(t), x(t), y(t), u(t), v_i, p) = 0 \\ g^m(x(t), y(t), u(t), p) \le 0 \end{matrix}\right\} t \in [t_i, t_{i+1}] \\ Y_m \\ x(t_i^-) = x(t_i^+) \\ w^m(x(t), v_i) = 0, t \in [t_i, t_{i+1}] \end{bmatrix}$$

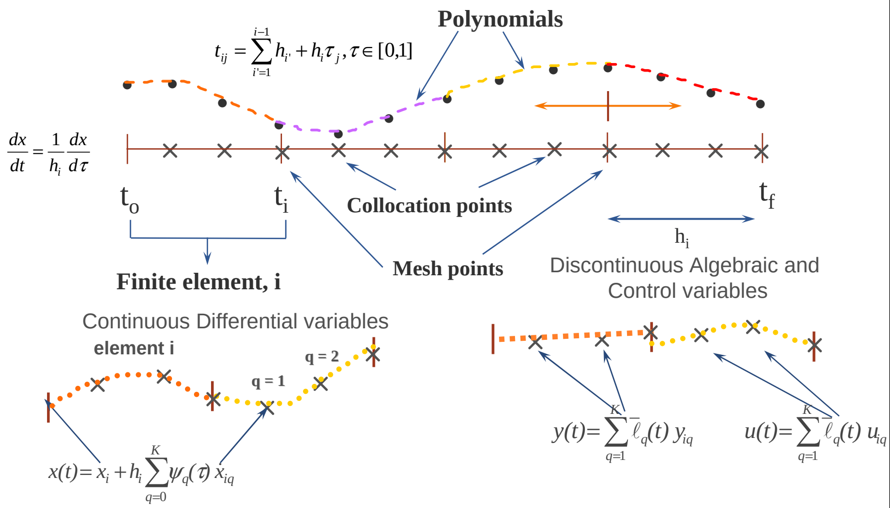
$$H(Y) = True,\ i = 0,\ldots N$$

Solution Strategy for Hybrid Problems

- Replace DAEs by algebraic approximations
- Represent disjunctions as complementarities at t_i
- Reformulate as MPEC and solve with NLP formulation

22

Modeling Epochs: Collocation on Finite Elements



$$\underline{F}(\dot{x}_{ik}, x_{ik}, y_{ik}, u_{ik}, p, \underline{v_i, h_i}) = 0, \quad k = 1,..K,\ i = 1,..\ N$$

- Piecewise constant switching profiles v_i, continuous if no resets
- High order ($2K$-1) IRK approximation to model epochs of variable length h_i

Converting Problem class to MPEC

$$\varsigma(x(t),u(t),t) > 0,\ \dot{x} = f_+(x,u,t)$$
$$\varsigma(x(t),u(t),t) < 0,\ \dot{x} = f_-(x,u,t)$$
$$\varsigma(x(t),u(t),t) = 0,\ \dot{x} = v(t) f_-(x,u) + (1 - v(t)) f_+(x,u), v(t) \in [0,1]$$
$$\Downarrow$$
$$\dot{x} = v^j f_-(x,u) + (1 - v^j) f_+(x,u), v^j \in [0,1]$$
$$\varsigma_j(x(t),u(t),t) = s_j^+ + s_j^-$$
$$0 \le s^+ \perp v \ \ge 0,\ 0 \le s^- \perp (1 - v\ ) \ge 0$$

But, need to enforce complementarity over entire epoch (element) i

$$\varsigma_j(x(t),u(t),t) = s_j^+ + s_j^-$$
$$0 \le \left\| s_j^+(t) \right\|_{t \in [t_i, t_{i+1})} \perp v_i^j \ge 0,\ 0 \le \left\| s_j^-(t) \right\|_{t \in [t_i, t_{i+1})} \perp (1 - v_i^j) \ge 0$$

Apply collocation over element i

$$\varsigma_j(x(t_{ik}),u(t_{ik}),t_{ik}) = s_{j,ik}^+ + s_{j,ik}^-$$
$$0 \le \sum_{k=0}^{K} s_{j,ik}^+ \perp v_i^j \ge 0,\ 0 \le \sum_{k=0}^{K} s_{j,ik}^- \perp (1 - v_i^j) \ge 0$$

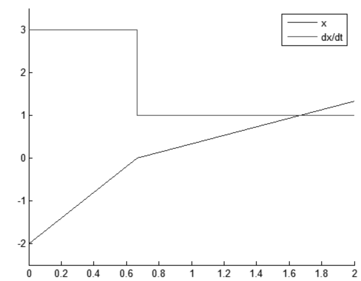
Apply penalty formulation to MPEC to convert to NLP

Example 1: Optimal control problem

adapted from Stewart, Anitescu (2005)

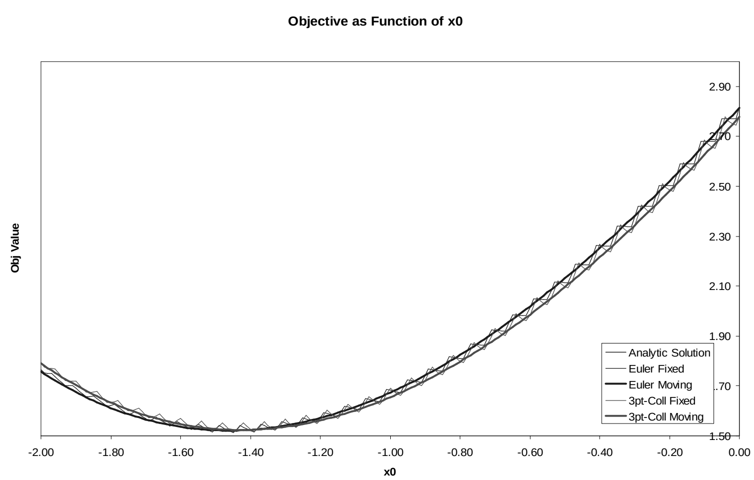
$$\min \quad (x_{end} - 5/3)^2 + \int_{t_0}^{t_{end}} x^2 dt$$

$$s.t. \quad x' = sgn(x) + 2, \; x(0) = -2$$



- Discretized with implicit Euler and solved with varying number of integration steps
- Apply moving finite elements with both Euler and Radau collocation
- Solved as both MPEC and MINLP

Objective Function vs. Initial Condition



Implicit Euler Method & 3-pt Collocation
100 finite elements
Moving vs. fixed finite elements

Comparison of MINLP & MPEC Formulations
(Implicit Euler)

	MINLP Formulation			MPEC Formulation		
	Objective	SBB	DICOPT	Objective	CONOPT	
NFE	Function	CPU sec	CPU sec	Function	Iterations	CPU sec
10	1.5364	0.266	1.841	1.6329	34	0.047
100	1.7581	69.125	274.257	1.7578	235	0.625
1000	1.7869	5555.058	>10000	1.7870	2365	47.967
2000	---	>10000	---	1.7882	4558	240.432
3000	---	---	---	1.7891	6253	475.753
4000	---	---	---	1.7893	8562	911.273

Example 2: Michael Schumacher Problem

Optimal control problem adapted from Stewart & Anitescu (2005)

- Problem initialized by solving with Euler's Method
- Final solution discretized with 3 point Radau collocation, 100 finite elements

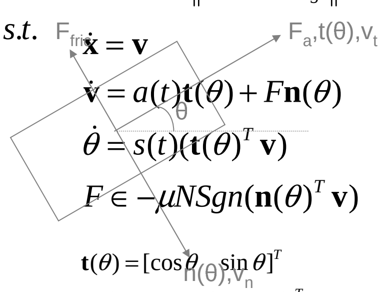
$$\min \quad T + 1000\left\|\mathbf{x}(T) - \mathbf{x}_{tgt}\right\|^2$$

$$s.t. \quad \dot{\mathbf{x}} = \mathbf{v}$$

$$\dot{\mathbf{v}} = a(t)\mathbf{t}(\theta) + F\mathbf{n}(\theta)$$

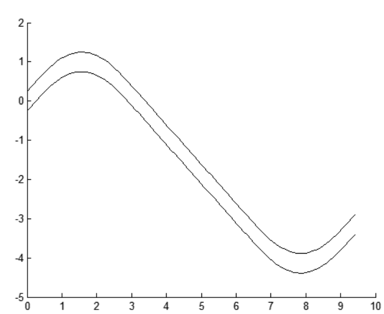
$$\dot{\theta} = s(t)(\mathbf{t}(\theta)^T \mathbf{v})$$

$$F \in -\mu N Sgn(\mathbf{n}(\theta)^T \mathbf{v})$$



$$\mathbf{t}(\theta) = [\cos\theta \quad \sin\theta]^T$$

$$\mathbf{n}(\theta) = [-\sin\theta \quad \cos\theta]^T$$



$$\theta(0) = 0$$

Schumacher Results 1

Original author's solution

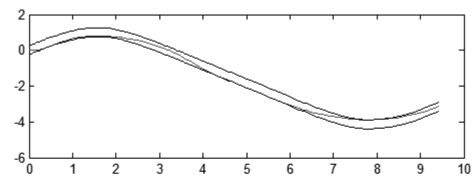
- T=5.541
- Obtained using Implicit Euler, 1000 fixed elements

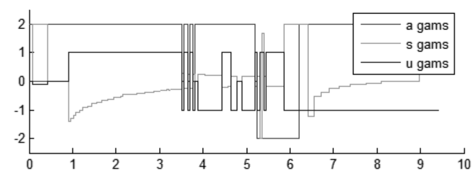
Our solution:

- T=4.773
- Error in final position (simulation of controls a,s,u):
  - $||x_{error}||_2$=2.25 x 10^{-4}

    $sgn(0) \Rightarrow u \in [-1,1]$
- Error in final position (simulation of controls a & s):
  - $||x_{error}||_2$=0.0555

    $sgn(0) \Rightarrow u = 0$





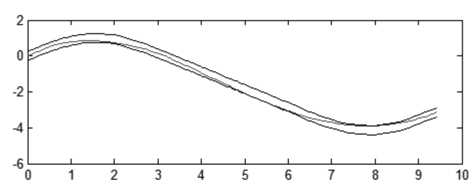
29

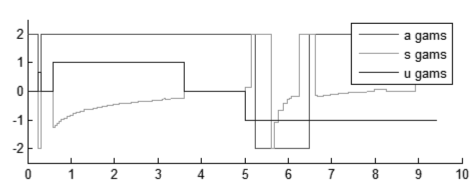
Schumacher Results 2

Modified Problem: $\theta(0) = \pi/4$

Solution:

- T=4.041
- Error(a,s,u):
  - $||x_{error}||_2$=2.49 x 10^{-6}
- Error(a,s):
  - $||x_{error}||_2$=0.0115





30

Industrial Gas Pipeline

Gas Network with multiple suppliers and customers

Challenges

- PDE constrained optimization
- Friction factor calculation
- Flow reversals

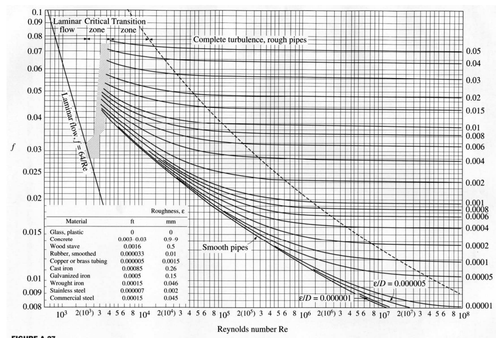
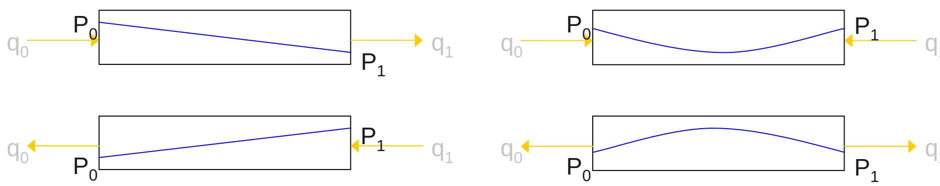


FIGURE A-27
The Moody chart for the friction factor for fully developed flow in circular tubes.



Pipe Segment Model

$$\frac{MW\cdot A_A L_A}{RT}\left(\frac{\overline{P}_{A,t+1}}{Z_{A,t+1}}-\frac{\overline{P}_{A,t}}{Z_{A,t}}\right)=\int_t^{t+1}\left(q_A^{in}-q_A^{out}\right)dt$$

$$\frac{\partial P}{\partial z}=\frac{-f\cdot ZRT}{2DA^2 MW}\frac{q|q|}{P}$$

$$\mathrm{Re}_{A,t}=\frac{|q_{A,t}|D_A}{\mu A_A}$$

$$|q_{A,t}|=q_{A,t}^{+}+q_{A,t}^{-}$$

$$q_{A,t}=q_{A,t}^{+}-q_{A,t}^{-}$$

$$0\le q_{A,t}^{+}\perp q_{A,t}^{-}\ge 0$$

$$\overline{P}_{A,t}=\frac{\int_{in}^{out}P_{A,t}\,dP}{L_A}$$

$$\overline{P}_{A,t}L_A A_A=q_A RT$$

$$f_{A,t}=SW_{A,t}f_{A,t}^{lam}+(1-SW_{A,t})f_{A,t}^{turb}$$

$$(\mathrm{Re}_{A,t}-2300)-\lambda_{A,t}^{1}+\lambda_{A,t}^{2}=0$$

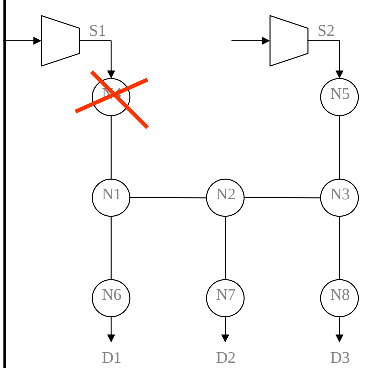
$$0\le SW_{A,t}\perp\lambda_{A,t}^{1}\ge 0$$

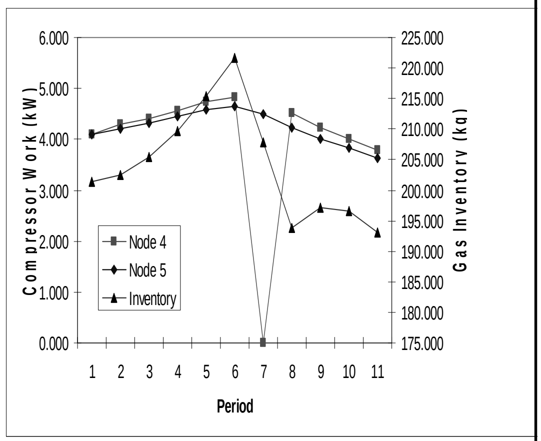
$$0\le(1-SW_{A,t})\perp\lambda_{A,t}^{2}\ge 0$$

$$f_{A,t}^{lam}=\frac{64}{\mathrm{Re}_{A,t}}$$

$$f_{A,t}^{turb}=1.326\left[Ln\left(\frac{1}{\frac{\varepsilon_A}{3.7D_A}+\frac{2.51}{\mathrm{Re}_{A,t}\sqrt{f_{A,t}^{turb}}}}\right)\right]^{-2}$$

Pipeline Packing Problem





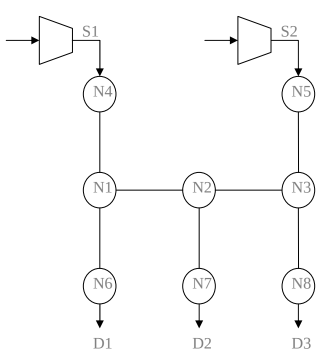
Model Attributes: 2609 Variables, 616 Complementarities
Solved with CONOPT: 3.645 CPU s, 145 Iterations

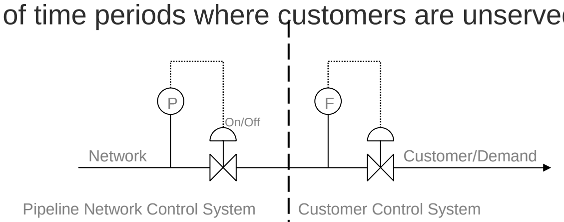
33

Minimum Recovery Time

What if Compressor Work is bounded [3.0,4.4]?

- Might not be feasible to serve all customers as desired
- Minimize number of time periods where customers are unserved





$$Min \sum_{d,t} v_{d,t}$$

s.t. pipeline models

$$q_{d,t} = q_{d,t}^{act} + q_{d,t}^{make}$$

$$q_{d,t}^{make} - \varepsilon = -\lambda^1_{d,t} + \lambda^2_{d,t}$$

$$0 \le \lambda^1_{d,t} \perp v_{d,t} \ge 0$$

$$0 \le \lambda^2_{d,t} \perp (1 - v_{d,t}) \ge 0$$

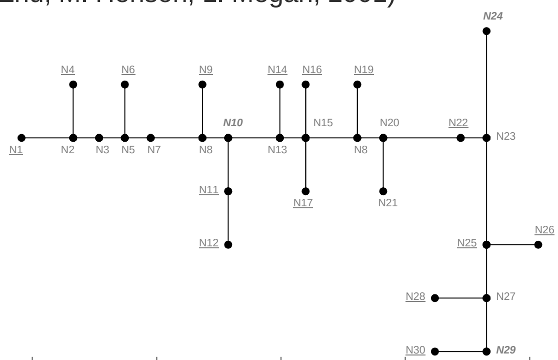
Unable to meet customer demand for D1 and D2 in time period 7.

34

Large Scale Example

Network from (G-Y. Zhu, M. Henson, L. Megan, 2001)

- 29 Arcs
- 30 Nodes
- 15 Customers
- 3 Suppliers



Time Periods	Iterations	Time (s)	Variables	Equations	MPEC Constraints
10	1092	397.953	14,858	14,621	2552
12	987	460.359	17,554	17,279	3016
18	2295	1633.766	25,642	25,253	4408
24	2917	3471.140	33,730	33,227	5800

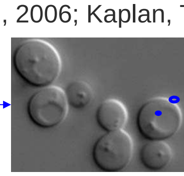
35

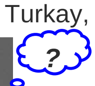
Parameter Estimation for Fermentation Processes

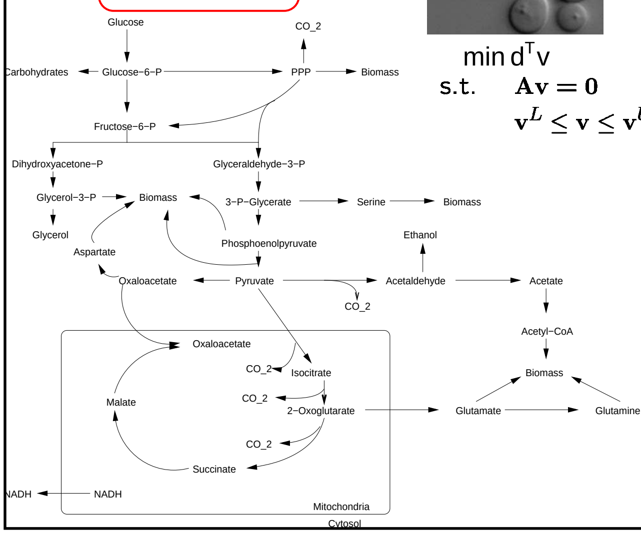
(Raghunathan, Perez, B., 2006; Kaplan, Turkay, Karasozen, B., 2008)

$$\mathbf{Av} = \mathbf{0}$$
$$\mathbf{v}^L \leq \mathbf{v} \leq \mathbf{v}^U$$

$\mathcal{S}$







min $d^T v$

s.t. $\mathbf{Av} = \mathbf{0}$

$\mathbf{v}^L \leq \mathbf{v} \leq \mathbf{v}^U$

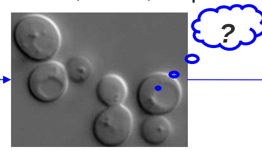
$$\frac{dc_i}{dt} = v_i^* c_{bio}$$

36

Parameter Estimation for Fermentation Processes

(Raghunathan et al., 2006; Kaplan et al., 2008)

$$\mathbf{A}\mathbf{v} = \mathbf{0}, \quad \mathbf{v}^L \le \mathbf{v} \le \mathbf{v}^U \;\rightarrow\; \mathcal{S}$$



$$\min\ \mathrm{d}^T\mathrm{v} \quad \text{s.t.} \quad \mathbf{A}\mathbf{v} = \mathbf{0}, \quad \mathbf{v}^L \le \mathbf{v} \le \mathbf{v}^U$$

$$\frac{\mathrm{d}c_i}{\mathrm{d}t} = v_i^* c_{bio}, \qquad \mathbf{c}(t_I) = \mathbf{c}_I$$

$$\mathbf{v}^* = \arg\min_{\mathbf{v}} \ \mathbf{d}^\top(\mathbf{c})\mathbf{v} \quad \text{s.t.} \quad \mathbf{A}\mathbf{v} = \mathbf{0}, \quad \mathbf{v}^L(\mathbf{c}) \le \mathbf{v} \le \mathbf{v}^U(\mathbf{c})$$

$\mathbf{c}_I$ → ODE → $\mathbf{c}(t)$; LP → $\mathbf{v}^*$

- •LP model for metabolic cellular network
- •Dynamics for evolution of fermentation products
- •NLP formulation for parameter estimation

- •Discretize DAE model in time
- • Substitute inner LP model by optimality and complementarity conditions
- •Solve as large single-level NLP

37

From Simulation to Optimization

$$\frac{\mathrm{d}c_i}{\mathrm{d}t} = v_i^* c_{bio}, \quad \mathbf{c}(t_I) = \mathbf{c}_I$$

$$v^* = \mathrm{argmin}\{d^T v \mid Av = 0, v^L \le v \le v^U\}$$

$\mathbf{c}_I$ → ODE → $\mathbf{c}(t)$; $\mathbf{v}^*$ LP

$$\min_z \ f(z) \quad \text{s.t.} \quad \text{constraints}, \quad \mu^U(\mathbf{v}^U - \mathbf{v}) \le 0, \quad \mu^L(\mathbf{v} - \mathbf{v}^L) \le 0$$

$$d(c_{NH_4}) = \begin{cases} d_1 & c_{NH_4} \ge \epsilon \\ d_2 & c_{NH_4} < \epsilon \end{cases} \qquad \min_u \ -u\left(c_{NH_4} - \epsilon\right) \quad \text{s.t.} \quad 0 \le u \le 1$$

$$d(c_{NH_4}) = u d_1 + (1-u) d_2$$

$$\min_{\mathbf{z}} \ f(\mathbf{z}) \quad \text{s.t.} \quad \mathbf{c}(\mathbf{z}) = \mathbf{0}, \quad Wy \le 0, \quad \mathbf{z}^L \le \mathbf{z} \le \mathbf{z}^U$$

constraints

$$c_i(t_k) = c_i(t_{k-1}) - \phi(v_i^* c_{bio})$$

$$\mathbf{d} + \mathbf{A}^\top \lambda - \mu^L + \mu^U = \mathbf{0}$$
$$\mathbf{A}\mathbf{v} = \mathbf{0}$$
$$\mathbf{v}^L \le \mathbf{v} \le \mathbf{v}^U$$
$$\{\mu^L, \mu^U\} \ge 0$$

complementarities

$$\mu^U(\mathbf{v}^U - \mathbf{v}) = \mathbf{0}$$
$$\mu^L(\mathbf{v} - \mathbf{v}^L) = \mathbf{0}$$

38

Parameter Estimation Results

$$\sum_{i \in \mathcal{M}_E} \theta_i z_i \rightarrow 1\text{gBiomass}$$

1150 rxn. → 38 rxn.
1062 met. → 42 met.

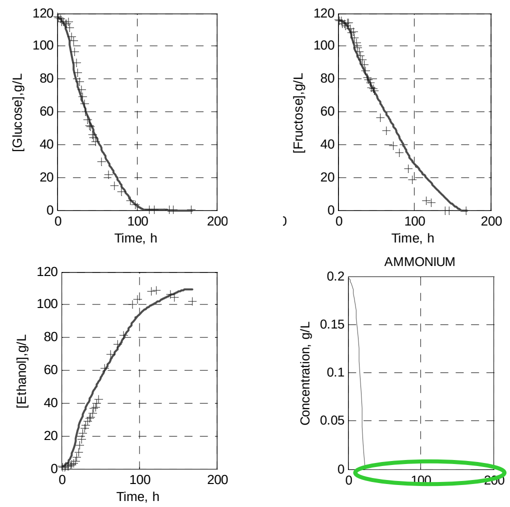
$$\min_{\theta} \sum_{i \in \mathcal{T}} \sum_{j \in \mathcal{M}_E} \left[c_j(t_i) - c_j^M(t_i)\right]^2$$

s.t. constraints

large-scale parameter estimation problem

VARIABLES	32,800
CONSTRAINTS	25,400
CPU-MINUTES	3.5

Compound *i*	mol *i*/gBio
Carbohydrate	1.54E-2
DNA	1.61E-4
RNA	1.65E-3
Lipids	1.46E-3
Protein	1.83E-2



Summary

Problems MPCCs can solve:

- Those involving inclusive disjunctions
  - (inclusive) OR constraints
- Those with piecewise functions
  - Abs(*), Min(*,*), Max(*,*), Signum(*), If * Then * Else *
- Hybrid Dynamic Systems
- Bilevel optimization problems

Other methods can solve these problems as well, but:

- MPCCs can be reformulated and solved with standard NLP solvers, with local solutions determined
- MPCCs can be solved quickly (polynomial time expected)