An Introduction to Complementarity

Michael C. Ferris

University of Wisconsin, Madison

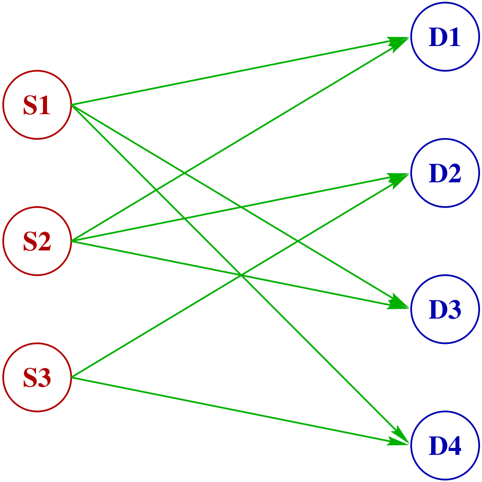
Nonsmooth Mechanics Meeting: June 14, 2010

Outline

- Introduction to Complementarity Models
- Extension to Variational Inequalities
- Extended Mathematical Programming
- Heirarchical Optimization

- Introduction: Transportation Model
- Application: World Dairy Market Model
- Algorithms: Feasible Descent Framework
- Implementation: PATH
- Results

Sample Network



Transportation Model

- Suppliers ship good from warehouses to customers
  - Satisfy demand for commodity
  - Minimize transportation cost
- Transportation network provided as set $\mathcal{A}$ of arcs
- Variables $x_{i,j}$ - amount shipped over $(i,j) \in \mathcal{A}$
- Parameters
  - s_i - supply at node i
  - d_i - demand at node i
  - $c_{i,j}$ - cost to ship good from nodes i to j

Linear Program

$$
\begin{array}{rlll}
\min_{x \geq 0} & \sum_{(i,j) \in \mathcal{A}} c_{i,j} x_{i,j} & & \\
\text{subject to} & \sum_{j:(i,j) \in \mathcal{A}} x_{i,j} \leq s_i & \forall\, i \\
& \sum_{i:(i,j) \in \mathcal{A}} x_{i,j} \geq d_j & \forall\, j
\end{array}
$$

Multipliers

- Introduce multipliers (marginal prices) p^s and p^d

$$\sum_{j:(i,j)\in\mathcal{A}} x_{i,j} \le s_i \qquad p_i^s \ge 0$$

- In a competitive marketplace

$$\sum_{j:(i,j)\in\mathcal{A}} x_{i,j} < s_i \quad \Rightarrow \quad p_i^s = 0$$

- At solution

$$\sum_{j:(i,j)\in\mathcal{A}} x_{i,j} = s_i \quad \text{or} \quad p_i^s = 0$$

- Complementarity relationship

$$\sum_{j:(i,j)\in\mathcal{A}} x_{i,j} \le s_i \quad \perp \quad p_i^s \ge 0$$

Wardropian Equilibrium

- Delivery cost exceeds market price

$$p_i^s + c_{i,j} \geq p_j^d$$

- Strict inequality implies no shipment $x_{i,j} = 0$
- Linear complementarity problem

$$\begin{array}{rcll} \sum_{j:(i,j)\in\mathcal{A}} x_{i,j} \leq s_i & \perp & p_i^s \geq 0 & \forall\, i \\ d_j \leq \sum_{i:(i,j)\in\mathcal{A}} x_{i,j} & \perp & p_i^d \geq 0 & \forall\, j \\ p_j^d \leq p_i^s + c_{i,j} & \perp & x_{i,j} \geq 0 & \forall\, (i,j) \in \mathcal{A} \end{array}$$

- First order conditions for linear program

Nonlinear Complementarity Problems

- Given $F : \Re^n \to \Re^n$
- Find $x \in \Re^n$ such that

$$0 \leq F(x) \qquad x \geq 0$$

$$x^T F(x) = 0$$

- Compactly written

$$0 \leq F(x) \quad \perp \quad x \geq 0$$

- Equivalent to nonsmooth equation:

$$\min(x, F(x)) = 0$$

Extensions

- Original problem has fixed demand
- Use general demand function $d(p)$
- Examples
  - Linear demand

$$\sum_{i:(i,j)\in\mathcal{A}} x_{i,j} \geq d_j(1 - p_j^d)$$

  - Nonlinear demand
    - Cobb-Douglas
    - CES function
- Use more general cost functions $c(x)$

Taxes and Tariffs

- Exogenous supply tax t_i

$$p_i^s(1 + t_i) + c_{i,j} \geq p_j^d$$

- Endogenous taxes
  - Make t_i a variable
  - Add driving equation
- No longer optimality conditions

Most complementarity problems do not correspond to first order conditions of optimization problems

Use of complementarity

- Pricing electricity markets and options
- Contact Problems with Friction
- Video games: model contact problems
  - Friction only occurs if bodies are in contact
- Crack Propagation
- Structure design
  - how springy is concrete
  - optimal sailboat rig design
- Congestion in Networks
- Computer/traffic networks
  - The price of anarchy measures difference between "system optimal" (MPCC) and "individual optimization" (MCP)
- Electricity Market Deregulation
- Game Theory (Nash Equilibria)
- General Equilibria with Incomplete Markets
- Impact of Environmental Policy Reform

World Dairy Market Model

- Spatial equilibrium model of world dairy sector
  - 5 farm milk types
  - 8 processed goods
  - 21 regions
- Regions trade raw and processed goods
- Barriers to free trade
  - Import policies: quotas, tariffs
  - Export policies: subsidies
- Study impact of policy decisions
  - GATT/URAA
  - Future trade negotiations

Formulation

- Quadratic program
  - Variables: quantities
  - Constraints: production and transportation
  - Objective: maximize net social welfare
- Difficulty is *ad valorem* tariffs
  - Tariff based on value of goods
  - Market value is multiplier on constraint
- Complementarity problem
  - Formulate optimality conditions
  - Market price is now a variable
  - Directly model *ad valorem* tariffs

World Dairy Market Model Statistics

- Quadratic program
  - 31,772 variables
  - 14,118 constraints
- Linear complementarity problem
  - 45,890 variables and constraints
  - 131,831 nonzeros

European Put Option

- Contract where holder *can* sell an asset
  1. At a fixed expiration time T
  2. For a fixed price E
- Asset value $S(t)$ satisfies

$$dS = (\sigma \, dX + \mu \, dt)S$$

  - $\sigma \, dX$ is random return
  - $\mu \, dt$ is deterministic return
- Black-Scholes Equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

American Put Options

- Contract where holder *can* sell an asset
  1. At any time prior to a fixed expiration T
  2. For a fixed price E
- Free Boundary $S_f(t)$ – optimal exercise price

$$V(S_f(t), t) = \max(E - S_f(t), 0)$$

$$\frac{\partial V}{\partial S}(S_f(t), t) = -1$$

- If $S \geq S_f(t)$ then satisfy Black-Scholes Equation

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0$$

Complementarity Problem

- Reformulate to remove dependence on boundary
- Partial differential complementarity problem

$$F(S,t) \equiv \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV$$

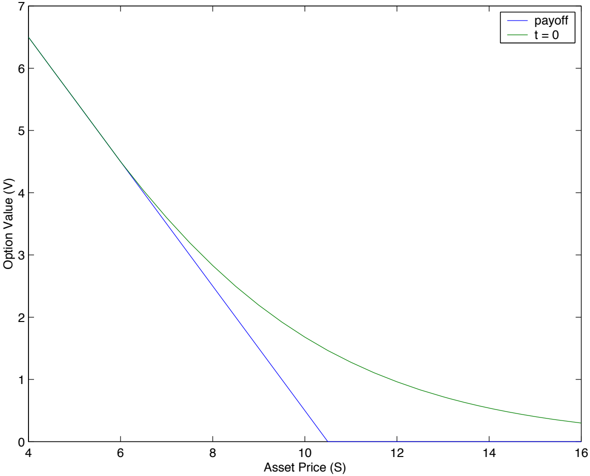
$$0 \leq -F(S,t) \quad \perp \quad V(S,t) \geq \max(E-S,0)$$

- Free boundary recovered after solving

Solution Method

- Finite differences used to discretize
  - Central differences for space
  - Forward differences for time
  - Crank-Nicolson method
- Step through time from T to present
- At each t a linear complementarity problem is solved
- Used GAMS/PATH to model and solve

Results



payoff
t = 0
Option Value (V)
Asset Price (S)
0
1
2
3
4
5
6
7
4
6
8
10
12
14
16

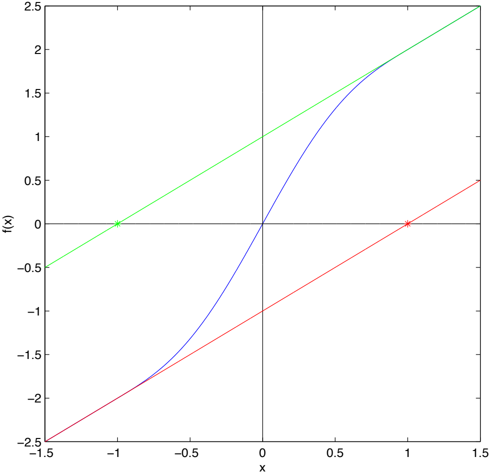
Background

- Nonlinear equations $F(x) = 0$
- Newton's Method

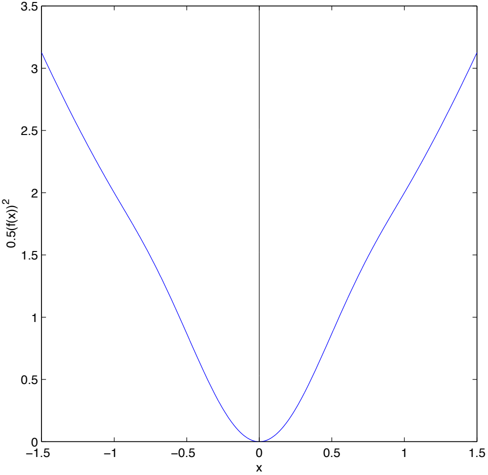
$$\begin{aligned} F(x^k) + \nabla F(x^k) d^k &= 0 \\ x^{k+1} &= x^k + d^k \end{aligned}$$

- Damp using Armijo linesearch on $\frac{1}{2} \|F(x)\|_2^2$
- Descent direction - gradient of merit function
- Properties
  - Well defined
  - Global and local-fast convergence

Cycling Example



Merit Function



Normal Map

- Equivalent piecewise smooth equation $F_+(x) = 0$

$$F_+(x) \equiv F(x_+) + x - x_+$$

- Nonsmooth Newton Method
  - Piecewise linear system of equations
  - Solve via a pivotal method
  - Damp using Armijo search on $\frac{1}{2} \|F_+(x)\|_2^2$
- Properties
  - Global and local-fast convergence
  - Merit function *not* differentiable

Nonlinear Complementarity Problem

- Assume $F_+(\bar{x}) = 0$
  1. If $\bar{x}_i \geq 0$ then $\bar{x}_i - (\bar{x}_i)_+ = 0$ and $F_i(\bar{x}) = 0$
  2. If $\bar{x}_i \leq 0$ then $\bar{x}_i - (\bar{x}_i)_+ \leq 0$ and $F_i(\bar{x}) \geq 0$
- Therefore $\bar{x}_+$ solves

$$0 \leq F(x) \quad x \geq 0$$

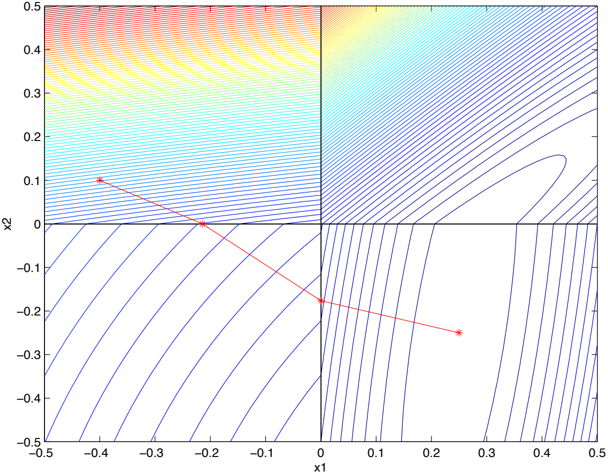
$$x^T F(x) = 0$$

- Compact representation

$$0 \leq F(x) \quad \perp \quad x \geq 0$$

- If $\bar{z}$ solves NCP(F) then $F_+(\bar{z} - F(\bar{z})) = 0$

Piecewise Linear Example



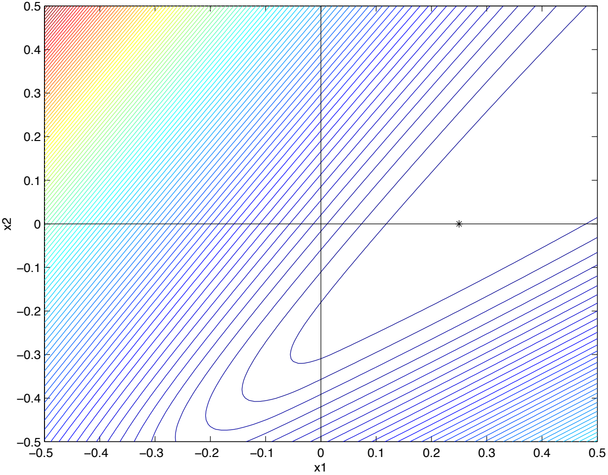
Fischer-Burmeister Function

- $\Phi(x)$ defined componentwise

$$\Phi_i(x) \equiv \sqrt{(x_i)^2 + (F_i(x))^2} - x_i - F_i(x)$$

- $\Phi(x) = 0$ if and only if x solves NCP(F)
- Not continuously differentiable - semismooth
- Natural merit function ($\frac{1}{2} \|\Phi(x)\|_2^2$) is differentiable

Fischer-Burmeister Example



Review

- Nonlinear Complementarity Problem
- Piecewise smooth system of equations
  - Use nonsmooth Newton Method
  - Solve linear complementarity problem per iteration
  - Merit function not differentiable
- Fischer-Burmeister
  - Differentiable merit function
- Combine to obtain new algorithm
  - Well defined
  - Global and local-fast convergence

Feasible Descent Framework

- Calculate direction using a local method
  - Generates feasible iterates
  - Local fast convergence
  - Used nonsmooth Newton Method
- Accept direction if descent for $\frac{1}{2}\|\Phi(x)\|^2$
- Otherwise use projected gradient step

Theorem

Let $\{x^k\} \subseteq \Re^n$ be a sequence generated by the algorithm that has an accumulation point x^* which is a strongly regular solution of the NCP. Then the entire sequence $\{x^k\}$ converges to this point, and the rate of convergence is Q-superlinear.

- Method is well defined
- Accumulation points are stationary points
- Locally projected gradient steps not used

Computational Details

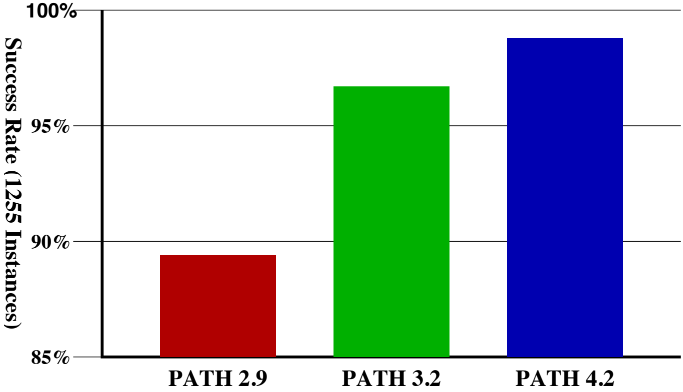
- Crashing method to quickly identify basis
- Nonmonotone search with watchdog
- Perturbation scheme for rank deficiency
- Stable interpolating pathsearch
- Restart strategy
- Projected gradient searches
- Diagnostic information

Restarts

- User provides solver with information
  - Starting point
  - Resource limits
- Effectively use resources to solve problem
- Determine when at a stationary point and $\|\Phi(x)\| > 0$
  - Restart from starting point
  - Modify algorithmic parameters
- Parameter choices based on empirical studies

Comparative Results

- Models from GAMSLIB and MCPLIB



Preprocessing

- Discover information about a problem
- Use to reduce size and complexity
  - Improve algorithm performance
  - Detect unsolvable models
- Main idea
  - Identify special structure
    - Polyhedral constraints
    - Separability
  - Use complementarity theory to eliminate variables

Linear Solve

- Majority of time spent finding direction
- Advanced starts
- Ill-conditioning and rank-deficiency
- Degeneracy in pivot sequence
- Cycling rules
- Stable regeneration of search path

Core Computation

- Each step solves a (large, sparse) linear system
- Pivot step updates system matrix by a rank-1 modification (see details later)
- Require factor, solve, update technology
  - Dense version: uses Fletcher Matthews updates of LU factors
  - Default version: uses LUSOL (Markovitz sparsity, Bartels Golub factor updates, rank revealing factorization)
  - New version: uses UMFPACK (unsymmetric multifrontal method, block LU updating (Schur Complement) for updates)
  - Compressed version: much more complicated to implement, not as efficient in practice over complete set of models

Availability

- Modeling Languages
  - GAMS
  - AMPL
- MATLAB
  - MEX interface
- NEOS
  - FORTRAN specification
  - ADIFOR to obtain Jacobian
  - Large problems solved via CONDOR
- Callable Library

Conclusions

- Complementarity generalizes nonlinear equations
- Nonsmooth Newton method proposed
  - Differentiable merit function
  - Well defined
  - Global and local-fast convergence
- Developed sophisticated implementation
- Applied to several problems
  - Transportation model
  - Options pricing
- Future – improve speed and reliability

Discussion

- Very robust on standard test set
- Obtained large, difficult models from colleagues
  - World Dairy Market Model
  - Several quadratic programs
- Improve performance on large scale problems
  - Robustness
  - Speed

Variational Inequality Formulation

- $F : \Re^n \to \Re^n$
- Ideally: $X \subseteq \Re^n$ – constraint set
- In practice: $X \subseteq \Re^n$ – simple bounds

$$0 \in F(x) + N_X(x)$$

- Special Cases
  - Nonlinear Equations ($X \equiv \Re^n$)

$$F(x) = 0$$

  - Nonlinear Complementarity Problem ($X \equiv \Re^n_+$)

$$0 \leq F(x) \quad x \geq 0$$
$$x^T F(x) = 0$$

Polyhedral Constraints X

- X and $N_X(\cdot)$ are geometric objects
- Free to choose algebraic representation
- Partition into two components: $X \equiv B \cap C$
  - B - simple bounds – treated specially by algorithm
  - C - polyhedral set
- Reduce complexity of C
- Must find X automatically

Polyhedral Structure

- Partition variables into (x, y)
- Identify skew symmetric structure

$$0 \in \begin{bmatrix} F(x) - A^T y \\ Ax - b \end{bmatrix} + \begin{bmatrix} N_{\Re^n_+}(x) \\ N_{\Re^n_+}(y) \end{bmatrix}$$

- Equivalent polyhedral problem (Robinson)

$$0 \in F(x) + N_{\Re^n_+ \cap \{x \mid Ax - b \geq 0\}}(x)$$

- Implementation finds a single constraint at a time

Relationship

1. **If $(\bar{x}, \bar{y})$ solves box constrained problem then $\bar{x}$ solves the polyhedral problem**
2. **If $\bar{x}$ solves the polyhedral problem then there exist multipliers $\bar{y}$ such that $(\bar{x}, \bar{y})$ solves the box constrained problem**

How do we calculate the multipliers, $\bar{y}$?

Calculating Multipliers

- Given an $\bar{x}$ solving the polyhedral problem
- Choose $\bar{y}$ solving the following linear program

$$\begin{array}{rc} \min_{y \in \Re^n_+} & y^T(A\bar{x} - b) \\ \text{s.t.} & 0 \in F(\bar{x}) - A^T y + N_{\Re^n_+}(\bar{x}) \end{array}$$

If $\bar{x}$ solves the polyhedral problem then

1. **The linear program is solvable**
2. **Given any $\bar{y}$ in the solution set, $(\bar{x}, \bar{y})$ solves the box constrained problem**

Separable Structure

- Partition variables into (x, y)
- Identify separable structure

$$0 \in \begin{bmatrix} F(x) \\ G(x,y) \end{bmatrix} + \begin{bmatrix} N_{\Re^n_+}(x) \\ N_{\Re^n_+}(y) \end{bmatrix}$$

- Reductions possible if either
  1. $0 \in F(x) + N_{\Re^n_+}(x)$ has a unique solution
  2. $0 \in G(x,y) + N_{\Re^n_+}(y)$ has solution for all x
- Theory provides appropriate conditions
- Solve F and G sequentially

Presolve

1. Identify a constraint with skew symmetric property
2. Convert problem into polyhedral form
3. Modify representation of polyhedral set
   - Singleton and doubleton rows
   - Forcing constraints
   - Duplicate rows
4. Recover box constrained problem with reduced size
   - Multipliers fixed and function modified
   - Additional polyhedral constraints uncovered
5. Repeat 1–4 until no changes
6. Identify separable structure

Example

- Original problem

$$0 \in \begin{bmatrix} x^2 - y - 1 \\ x - 1 \end{bmatrix} + \begin{bmatrix} N_{\Re_+}(x) \\ N_{\Re_+}(y) \end{bmatrix}$$

- Polyhedral problem

$$0 \in x^2 - 1 + N_{\Re_+ \cap \{x | x - 1 \geq 0\}}(x)$$

- Equivalent problem

$$0 \in x^2 - 1 + N_{[1,\infty)}(x)$$

Example (continued)

- $0 \in x^2 - 1 + N_{[1,\infty)}(x)$ has one solution $\bar{x} = 1$
- Solve optimization problem

$$\begin{array}{cc} \min_{y \in \Re_+} & y^T(\bar{x} - 1) \\ \text{s.t.} & 0 \in \bar{x}^2 - y - 1 + N_{\Re_+}(\bar{x}) \end{array}$$

- Equivalent model

$$\begin{array}{cc} \min_{y \in \Re_+} & 0 \\ \text{s.t.} & y = 0 \end{array}$$

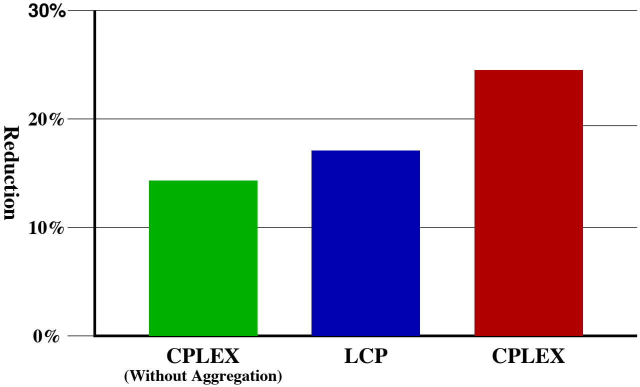
- Obtain $\bar{y} = 0$
- Solution is $(1, 0)$

Availability of Preprocessor

- PATH 4.3 for GAMS and AMPL
  - Finds polyhedral structure
  - Exploits separable structure
- Capability exists in other environments
  - User needs to provide information
  - Listing of linear/nonlinear elements in Jacobian
  - Optional - interval evaluation routines

Results: Linear Programs

- Formulated first order conditions of NETLIB problems
- Polyhedral structure *not* supplied to LCP preprocessor



Results: Quadratic Programs

- Solve optimality conditions
- Synthetic models
  - NETLIB problems with $\frac{1}{2}\|x\|^2$ added to objective
  - Selected 8 models
  - 17.6% reduction in size
  - 29.2% reduction in time
- World Dairy Market Model
  - Failed on original model (4.5 hours)
  - 70.4% reduction in size
  - Solved preprocessed model 23 minutes
  - 91.5% reduction in time!

Results: Nonlinear Complementarity Problems

- Models from GAMSLIB and MCPLIB
- Selected 6 models
- 9.7% reduction in size
- 15.3% reduction in time

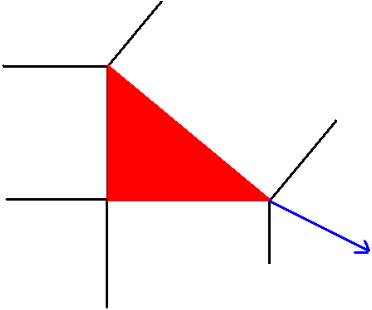
World Dairy Market Model Statistics

- Quadratic program
  - 31,772 variables
  - 14,118 constraints
- Linear complementarity problem
  - 45,890 variables and constraints
  - 131,831 nonzeros
- Preprocessed problem
  - 22,159 variables and constraints
  - 70,475 nonzeros

World Dairy Market Model Results

- Note: want to analyze large number of scenarios
- Gauss-Seidel method
  - Solves 96 quadratic programs
    - Uses MINOS with nonstandard options
  - Approximates equilibrium in 42 minutes
- Complementarity formulation
  - Solves a single complementarity problem
  - Computes equilibrium in
    - 117 minutes without preprocessing
    - 21 minutes with preprocessing
    - 11 minutes with nonstandard options
  - Obtain more accurate result in less time!

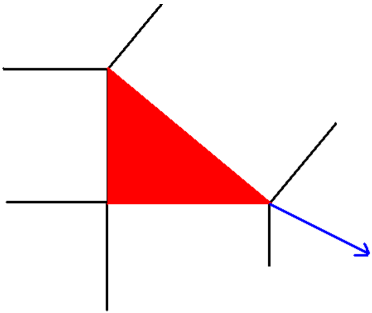
Normal map for polyhedral C



projection: $\pi_C(x)$

$$x - \pi_C(x) \in N_C(\pi_C(x))$$

Normal map for polyhedral C



projection: $\pi_C(x)$

$$x - \pi_C(x) \in N_C(\pi_C(x))$$

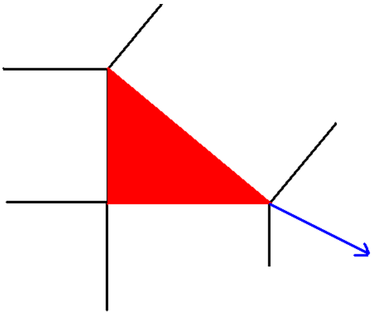
If $-M\pi_C(x) - q = x - \pi_C(x)$ then

$$-M\pi_C(x) - q \in N_C(\pi_C(x))$$

so $z = \pi_C(x)$ solves

$$\langle -Mz - q, y - z \rangle \leq 0, \quad \forall y \in C$$

Normal map for polyhedral C



projection: $\pi_C(x)$

$$x - \pi_C(x) \in N_C(\pi_C(x))$$

If $-M\pi_C(x) - q = x - \pi_C(x)$ then

$$-M\pi_C(x) - q \in N_C(\pi_C(x))$$

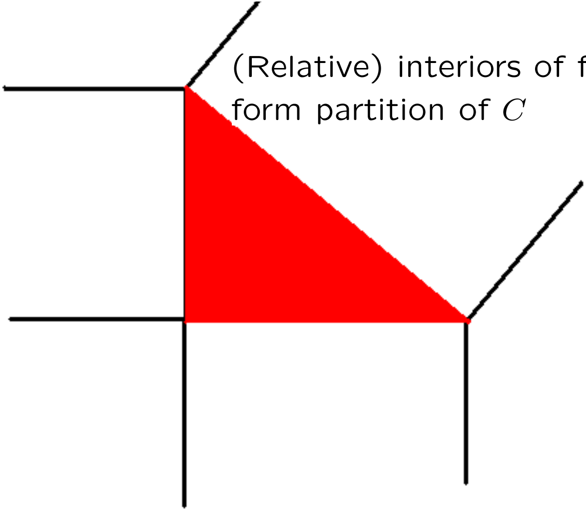
so $z = \pi_C(x)$ solves

$$\langle -Mz - q, y - z \rangle \leq 0, \quad \forall y \in C$$

Find x, a zero of the normal map:

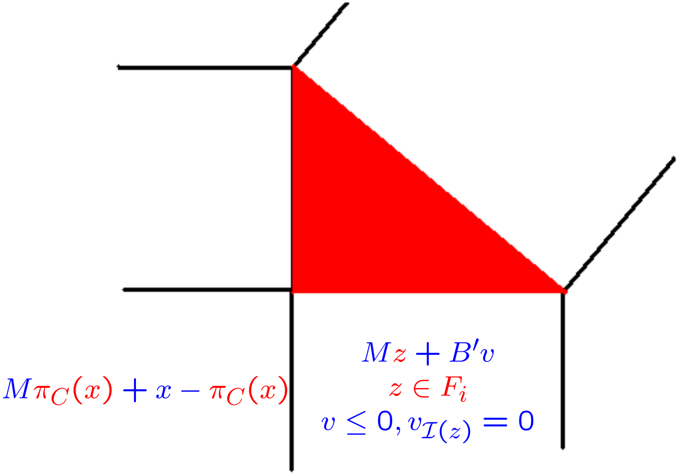
$$0 = M\pi_C(x) + q + x - \pi_C(x)$$

Normal manifold $= \{F_i + N_{F_i}\}$

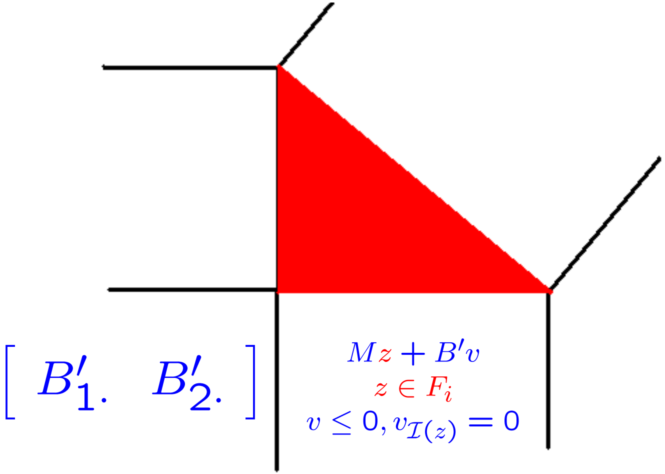


(Relative) interiors of faces F_i form partition of C

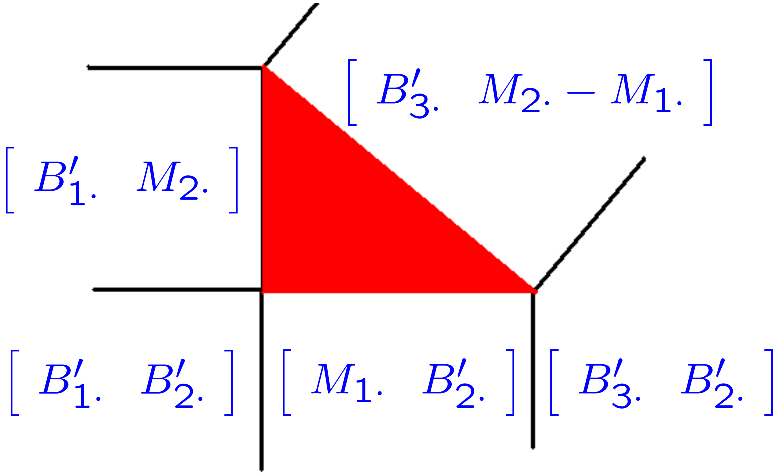
$$C = \{z | Bz \geq b\}, N_C(z) = \{B'v | v \leq 0, v_{\mathcal{I}(z)} = 0\}$$



$$C = \{z | Bz \geq b\}, N_C(z) = \{B'v | v \leq 0, v_{\mathcal{I}(z)} = 0\}$$

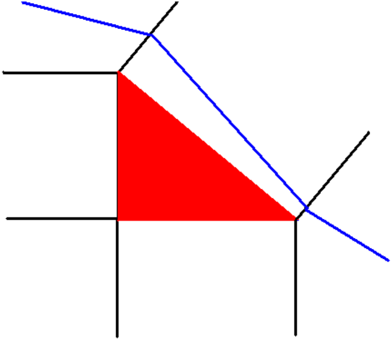


$$C = \{z | Bz \geq b\}, N_C(z) = \{B'v | v \leq 0, v_{\mathcal{I}(z)} = 0\}$$



Cao/Ferris Path (Eaves)

- Start in cell that has interior (face is an extreme point)
- Move towards a zero of affine map in cell
- Update direction when hit boundary (pivot)
- Solves or determines infeasible if M is copositive-plus on rec(C)
- Solves 2-person bimatrix games, 3-person games too, but these are nonlinear



But algorithm has exponential complexity (von Stengel et al)

Extensions and Computational Results

- Embed AVI solver in a Newton Method - each Newton step solves an AVI
- Compare performance of PathAVI with PATH on equivalent LCP
- PATH the most widely used code for solving MCP
- AVIs constructed to have solution with $M_{n\times n}$ symmetric indefinite

	PathAVI		*PATH*	
Size (m,n)	*Resid*	*Iter*	*Resid*	*Iter*
$(180, 60)$	3×10^{-14}	193	0.9	10176
$(360, 120)$	3×10^{-14}	1516	2.0	10594

- 2 - 10x speedup in Matlab using sparse LU instead of QR
- 2 - 10x speedup in C using sparse LU updates

Conclusions

- Complementarity problems abound in multiple application domains
- The PATH solver is a large scale (black-box) implementation of a (nonsmooth) Newton method for solving complementarity problems
- The PATH solver is available for download at http://www.cs.wisc.edu/∼ferris/path.html
- Mathematically rigourous extensions to Variational Inequalities and specific structures in models possible

References

M. Cao and M. C. Ferris.
A pivotal method for affine variational inequalities.
Mathematics of Operations Research, 21:44–64, 1996.

M. C. Ferris, C. Kanzow, and T. S. Munson.
Feasible descent algorithms for mixed complementarity problems.
Mathematical Programming, 86:475–497, 1999.

M. C. Ferris and T. S. Munson.
Preprocessing complementarity problems.
In M. C. Ferris, O. L. Mangasarian, and J. S. Pang, editors, *Complementarity: Applications, Algorithms and Extensions*, volume 50 of *Applied Optimization*, pages 143–164, Dordrecht, The Netherlands, 2001. Kluwer Academic Publishers.

M. C. Ferris and J. S. Pang.
Engineering and economic applications of complementarity problems.
SIAM Review, 39:669–713, 1997.