fminimax

Stewart Curry, Kristy Mitchell,

Robert Smith, Jasmine West

Outline

- Introduction
- Activity
- Visualization
- Sequential Quadratic Programming Method
- MATLAB Examples
- Conclusion

Introduction

- fminimax solves constrained optimization problems of the following form:

$$
\min_{x} \max_{F_i} \{F_i(x)\} \text{ such that } \begin{aligned} c(x) &\leq 0 \\ ceq(x) &= 0 \\ Ax &\leq b \\ Aeq \cdot x &= beq \\ lb \leq x &\leq ub \end{aligned}
$$

Introduction

- The minimax problem is to minimize the maximum value of a set of multivariable functions $\{F_i(x)\}$
- This can be thought of as minimizing the worst case outcome.
- Starts at an initial estimate x_0
- Can solve constrained and unconstrained minimax problems

Kung Fu!!

Kung Fu Fighting

Punch	Chop	Eye Gouge
-1	0	1

It's time to fight! Find a partner and decide which of you is player 1 and which is player 2. You each start off with 20 health points. Pick an attack from above just like in rock paper scissors. You can find out how much damage you've done with the following formula, where P1 and P2 are the numbers given above for player 1's and player 2's attacks, respectively.

$$\begin{bmatrix} 1 & 2 \\ -2 & -1 \end{bmatrix} \begin{bmatrix} P1 \\ P2 \end{bmatrix} - 3 = \begin{bmatrix} DamageToPlayer1 \\ DamageToPlayer2 \end{bmatrix}$$

Or, to save time, just consult these charts:

Damage to Player 1				
	Player 2 Attack			
Player 1 Attack		-1	0	1
	-1	-6	-4	-2
	0	-5	-3	-1
	1	-4	-2	0

Damage to Player 2				
	Player 2 Attack			
Player 1 Attack		-1	0	1
	-1	0	-1	-2
	0	-2	-3	-4
	1	-4	-5	-6

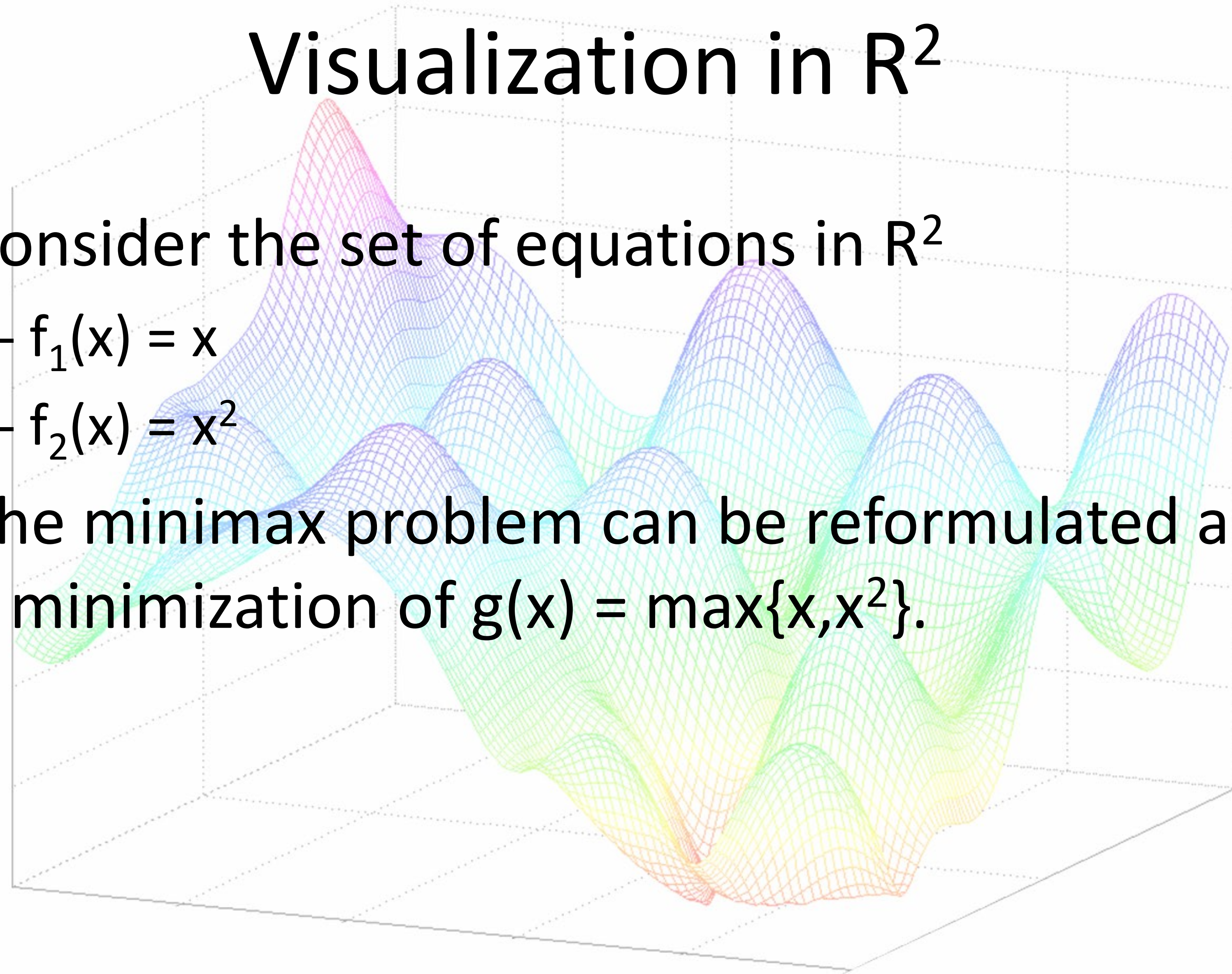
Keep track of your health total and play multiple rounds. When you're reduced to 0 or less, you're KO'd and lose the game!

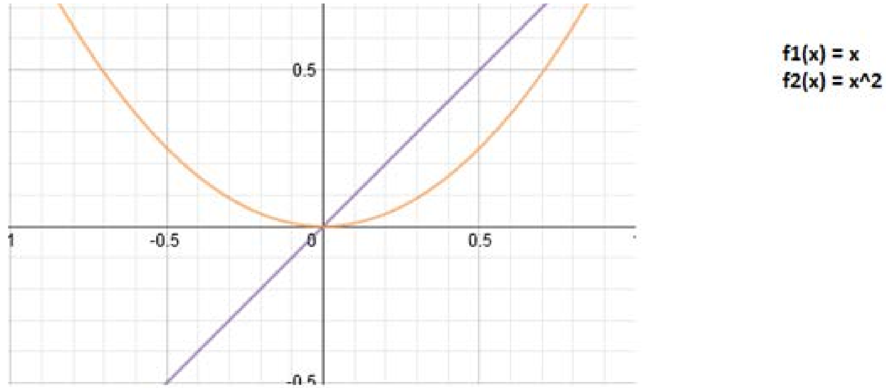
Solution

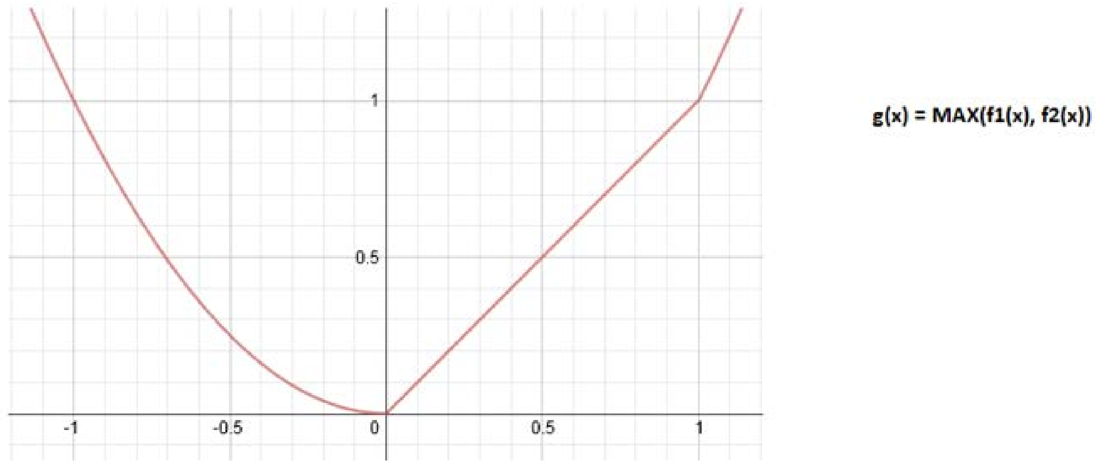
If you really want to clobber your opponent, one strategy is to choose the attack which will, in the worst case, deal as much damage as possible. For player one, that's Eye Gouge, since it will deal at least -4, while punch and chop might only deal -2 or 0, respectively. How did we arrive at this strategy? By selecting the one which minimized the maximum of the values in the outcome vector, which is a "minimax" problem.

Visualization in R^2

- Consider the set of equations in R^2
  - $f_1(x) = x$
  - $f_2(x) = x^2$
- The minimax problem can be reformulated as a minimization of $g(x) = \max\{x, x^2\}$.





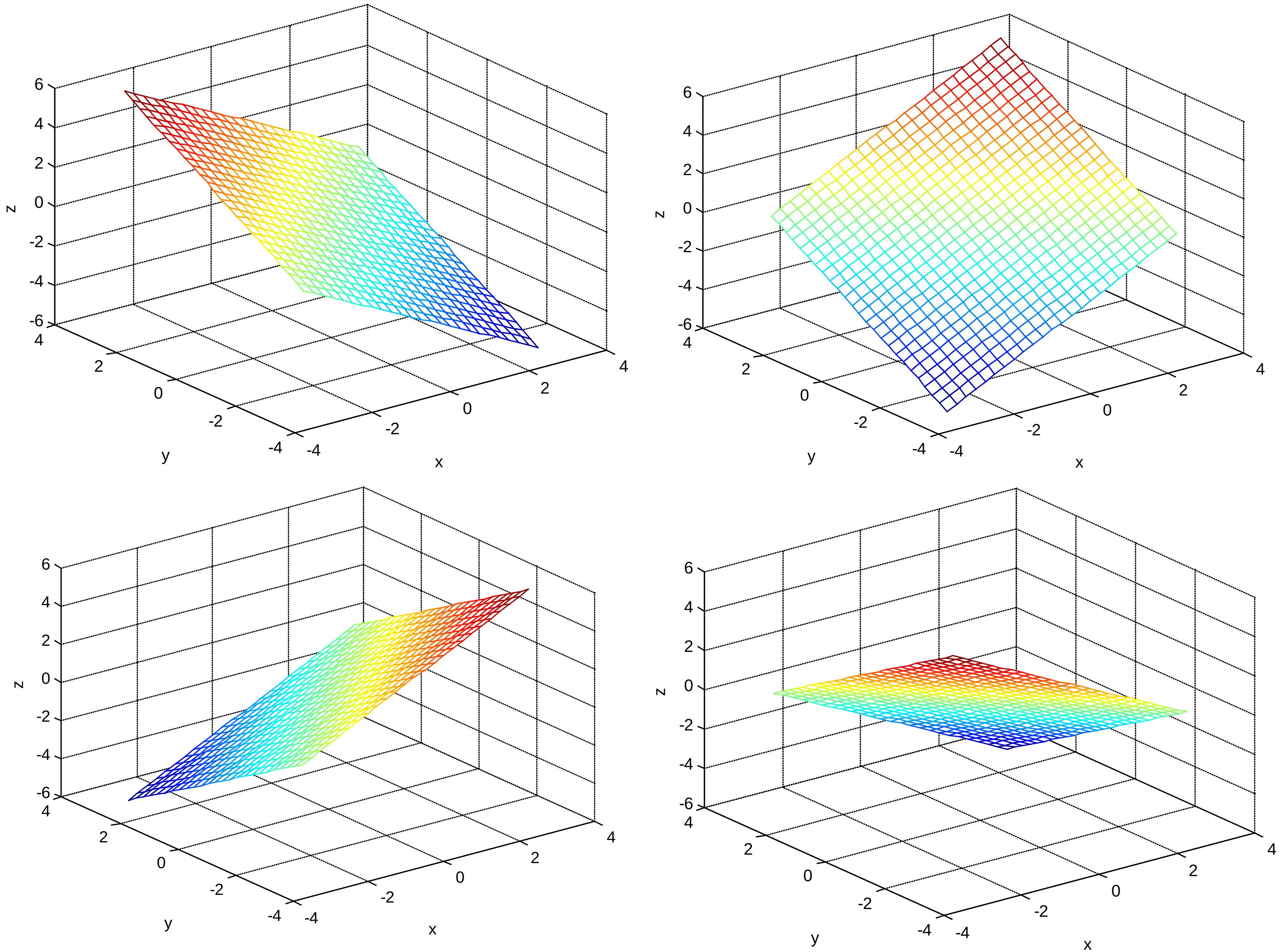


As you can see, the minimax solution for $f_1(x)$ and $f_2(x)$ is $x = 0$.

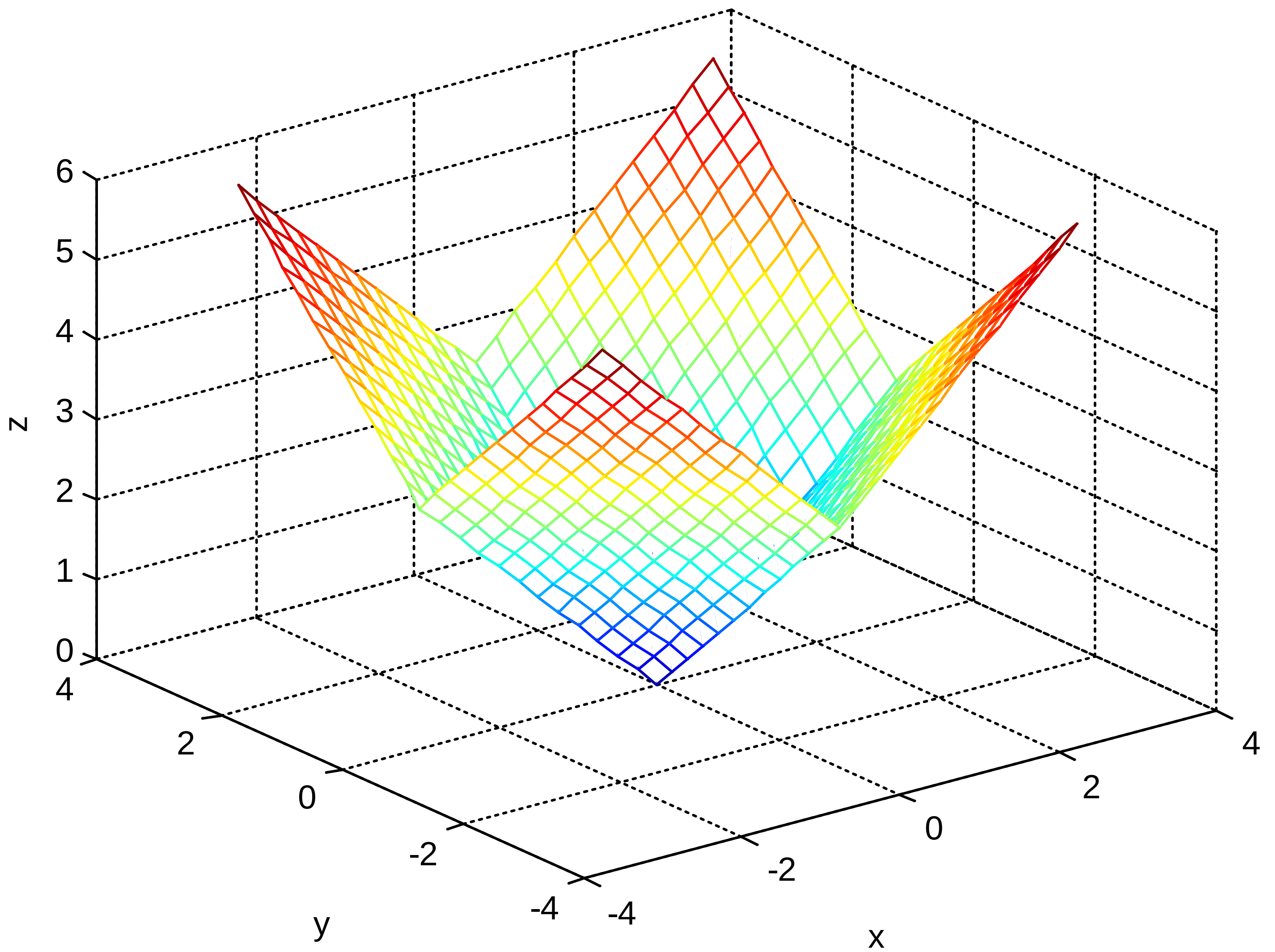
Visualization in R^3

- Now consider the following equations in R^3:
  - $f_1(x) = x + y$
  - $f_2(x) = x - y$
  - $f_3(x) = y - x$
  - $f_4(x) = -(x + y)$
- Again, the minimax problem can be thought of the minimization of

$$g(x) = \max\{f_1(x), f_2(x), f_3(x), f_4(x)\}$$



z
6
4
2
0
-2
-4
-6
y
4
2
0
-2
-4
x
-4
-2
0
2
4
z
6
4
2
0
-2
-4
-6
y
4
2
0
-2
-4
x
-4
-2
0
2
4
z
6
4
2
0
-2
-4
-6
y
4
2
0
-2
-4
x
-4
-2
0
2
4
z
6
4
2
0
-2
-4
-6
y
4
2
0
-2
-4
x
-4
-2
0
2
4



z
6
5
4
3
2
1
0
y
4
2
0
-2
-4
x
-4
-2
0
2
4

Sequential Quadratic Programming Method

Used by fminimax to solve problems

Iterative method used to solve nonlinear optimization problems

Three main stages of this algorithm in order to come to the solution

Objective functions and constraints must be twice continuously differentiable

Updating the Hessian Matrix

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k}$$

Positive definite quasi-Newton approximation of the Hessian

Value must remain positive at each update

$$s_k = x_{k+1} - x_k$$

$$q_k = \left(\nabla f(x_{k+1}) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x_{k+1}) \right) - \left(\nabla f(x_k) + \sum_{i=1}^{m} \lambda_i \nabla g_i(x_k) \right)$$

Each entry in this matrix updated term by term to ensure $q_k^T s_k$ remains positive.

Quadratic Programming Solution Subproblem

$$
\begin{aligned}
\min_{d \in \mathbb{R}^n} q(d) &= \frac{1}{2} d^T H d + C^T d \\
A_i d &= b_i, \;\; i = 1, \ldots, m_e \\
A_i d &\leq b_i, \;\; i = m_e + 1, \ldots, m
\end{aligned}
$$

Phase 1: Calculation of a feasible point (if one exists)

Phase 2: Generation of iterative sequence of feasible points that converge to the solution.

Line Search and Merit Function

With the solution we can now form a new iterate.

$$x_{k+1} = x_k + \alpha d_k$$

Solution to the Quadratic Programming Solution subproblem.

$$\alpha_k$$

Step Length Parameter which helps to create decrease in the Merit function

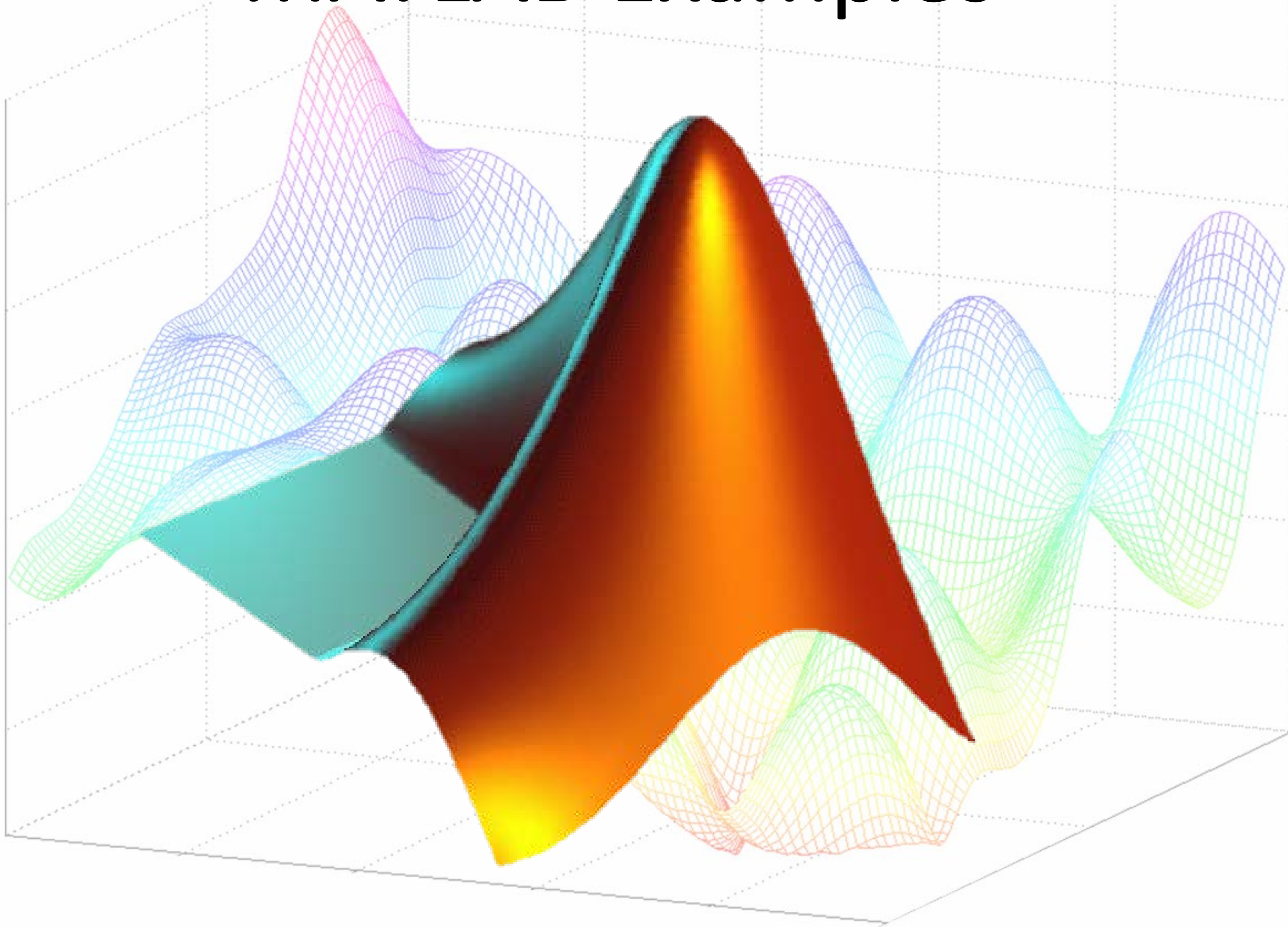
$$\Psi(x) = f(x) + \sum_{i=1}^{m_e} r_i g_i(x) + \sum_{i=m_e+1}^{m} r_i \cdot \max[0, g_i(x)]$$

Merit function

$$r_i = (r_{k+1})_i = \max_i \left\{ \lambda_i, \frac{(r_k)_i + \lambda_i}{2} \right\}, i = 1, \ldots, m$$

$$r_i = \frac{\|\nabla f(x)\|_2}{\|\nabla g_i(x)\|_2}$$

MATLAB Examples



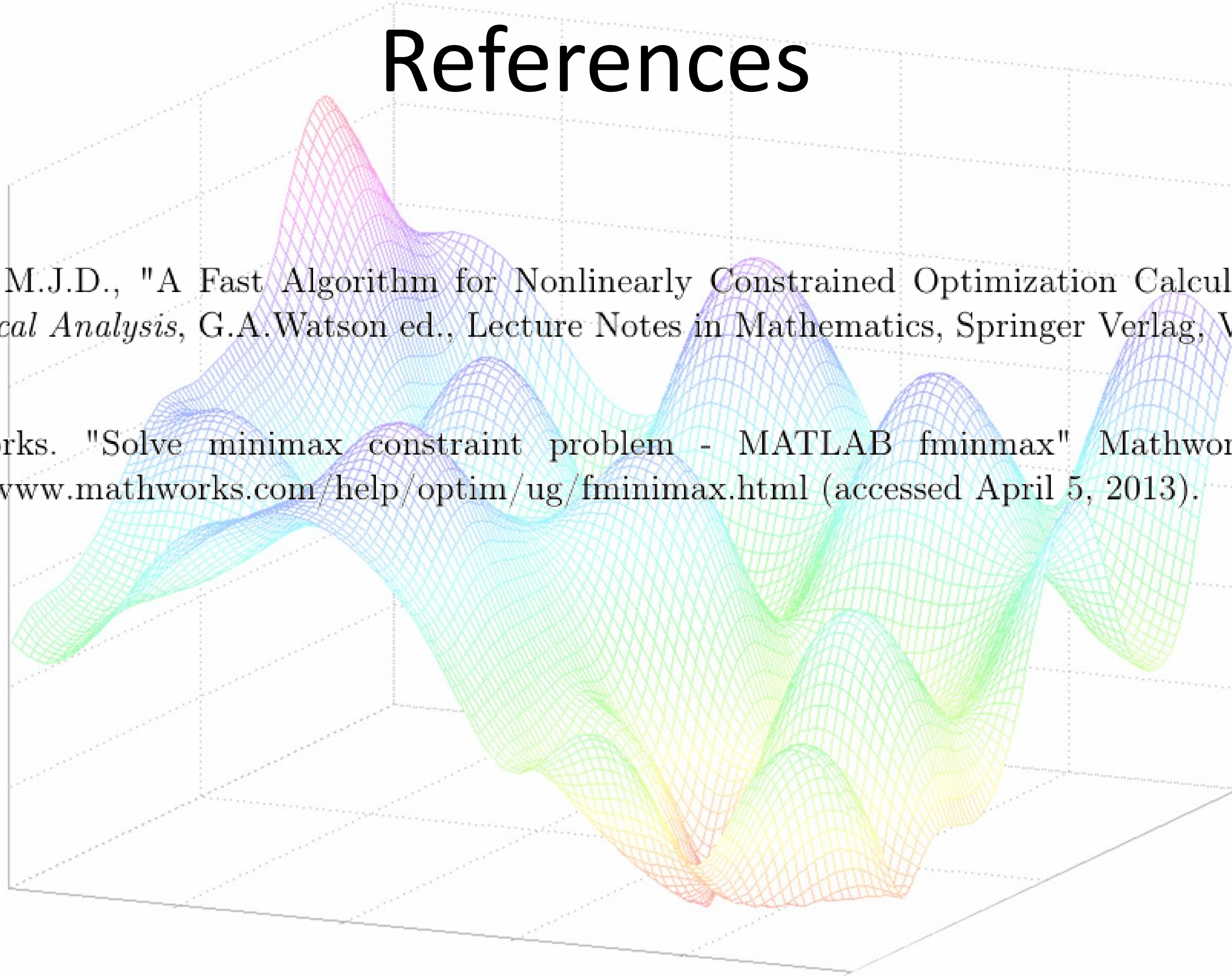
Conclusion

fminimax solves constrained and unconstrained minimax problems which seek to **minimize the largest value** for a set of multivariable functions

The algorithm accomplishes this task through a **sequential quadratic programming method** which solves the minimax program iteratively.

Overall, *fminimax*, is another tool useful for solving optimization problems.

References



[1] Powell, M.J.D., "A Fast Algorithm for Nonlinearly Constrained Optimization Calculations," *Numerical Analysis*, G.A.Watson ed., Lecture Notes in Mathematics, Springer Verlag, Vol. 630, 1978.

[2] Mathworks. "Solve minimax constraint problem - MATLAB fminmax" Mathworks.com. http://www.mathworks.com/help/optim/ug/fminimax.html (accessed April 5, 2013).