https://www.youtube.com/watch?v=Xs0NTNQmLYY

# Modern Optimization Techniques

Department of Electrical Engineering
College of Engineering
Salahaddin university-Erbil
***Prepared By: Diary R. SULAIAMAN***
diary.sulaiman@su.edu.krd
diary.sulaiman@gmail.com

***MSc Course***

***Second Semester***

***2020-2021***

Modern Computer Architecture

MSc Course

Second Semester

# CH1-CH2

# Review and Introduction



Prepared by: Diary R. Sulaiman

# Goals

- Learning classical and modern optimization techniques;
- acquiring an ability to apply to engineering problems;
- acquiring a broad understanding of the importance, scope, and current state of the optimization theory.

# Learning Outcomes

- An overall understanding of the importance of optimization in engineering as a ***mathematical tool***.
- Skills to implement classical optimization techniques in real engineering problems.
- An understanding of modern optimization techniques.
- Ability to use various software platforms involving optimization.
- An understanding of the limitations and applicability of particular optimization techniques.

# Introduction

- Optimization is the act of obtaining the best result under given conditions/circumstances (***Best Possible***)

- Optimization can be defined as the process of ***finding the conditions that give the maximum or minimum of a function***.

- The optimum seeking methods are also known as ***mathematical programming techniques*** and are generally studied as a part of ***operations research***.

- ***Operations research*** is a branch of mathematics concerned with the application of scientific ***methods and techniques*** to decision making problems and with establishing the best or optimal solutions.

# Applications

Some representative applications include:

- **Optimal routing in communication networks.**
- **Neural network training and applications in recent approximate dynamic programming techniques.**
- **Pattern recognition and classification.**
- **Optimal resource allocation in manufacturing and communication systems.**
- **Applications of semidefinite programming in combinatorial optimization, control theory, and design of chips.**
- **Estimation and system identification.**
- **Optimal control problems (e.g., rocket launching).**

# Course Content

- Introduction to optimization
  Classification of Optimization Algorithms
  What is an optimum?
    Single Objective Functions
    Multiple Objective Functions
    Constraint Handling
  The Structure of Optimization
  Problems in Optimization
- Classical Optimization: Constrained Multivariable Optimization
  Classical Optimization: Single-Variable Optimization, Unconstrained Multivariable Optimization Linear, quadratic, and geometric programming
- Theoretical Concepts of Nonlinear Programming
- Introduction to Modern Methods: GA, Swarm Opt., Ant Colony Opt.
- Hill Climbing
- …..

# Text Books

Springer www.springer.com

Mathematics - Applications | Modern Optimization Modelling Techniques

## Modern Optimization Modelling Techniques

Series: » Advanced Courses in Mathematics - CRM Barcelona

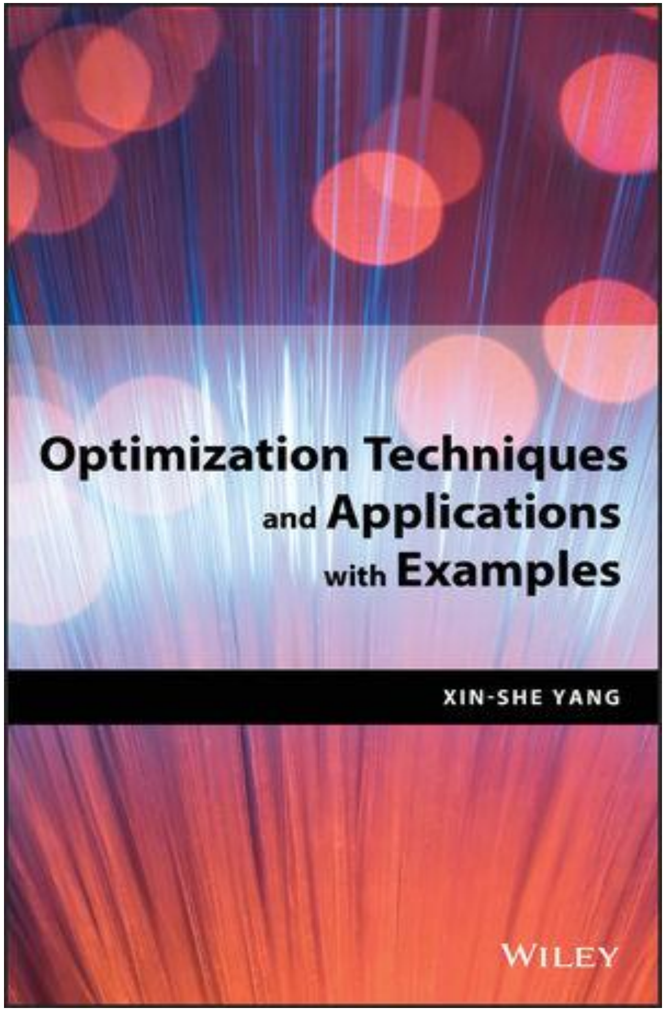
**Cominetti**, Roberto, **Facchinei**, Francisco, **Lasserre**, Jean B.

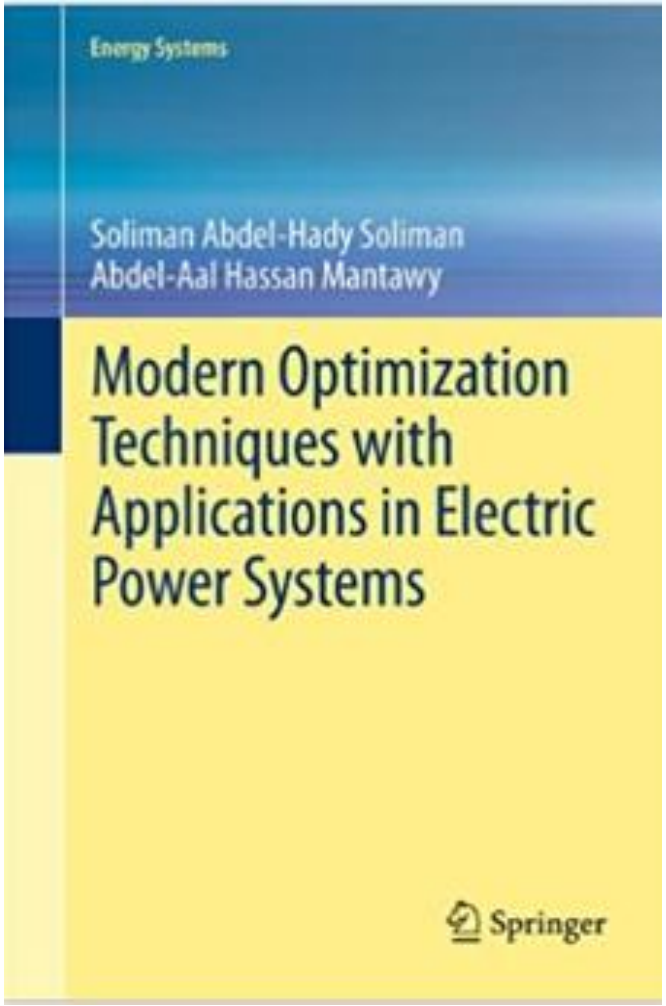
1st Edition., 2012, Approx. 290 p.

A product of Springer Basel

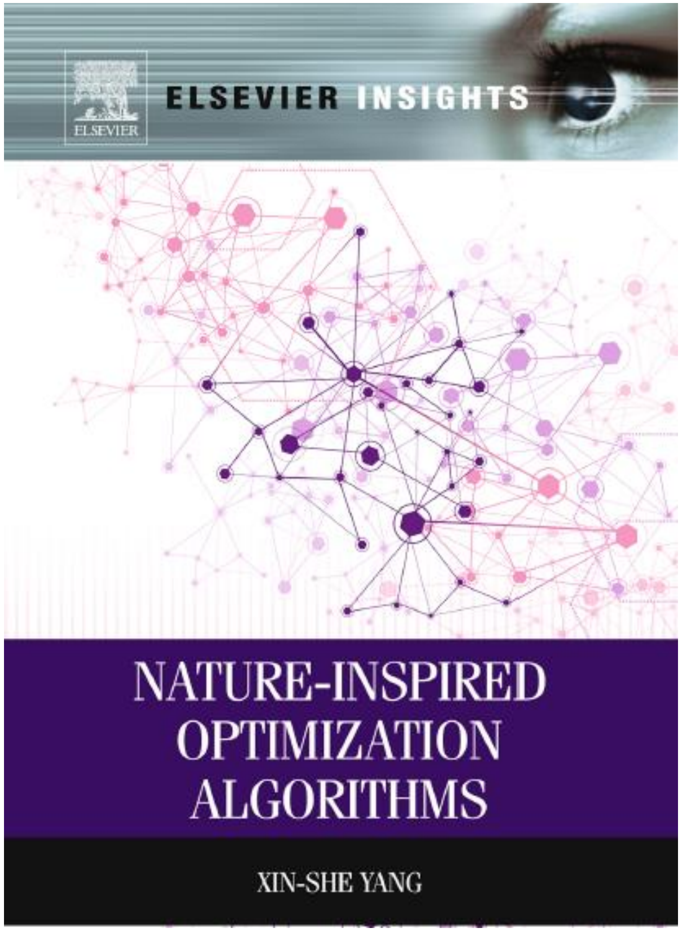
**Softcover**, ISBN 978-3-0348-0290-1
Due: July 29, 2012

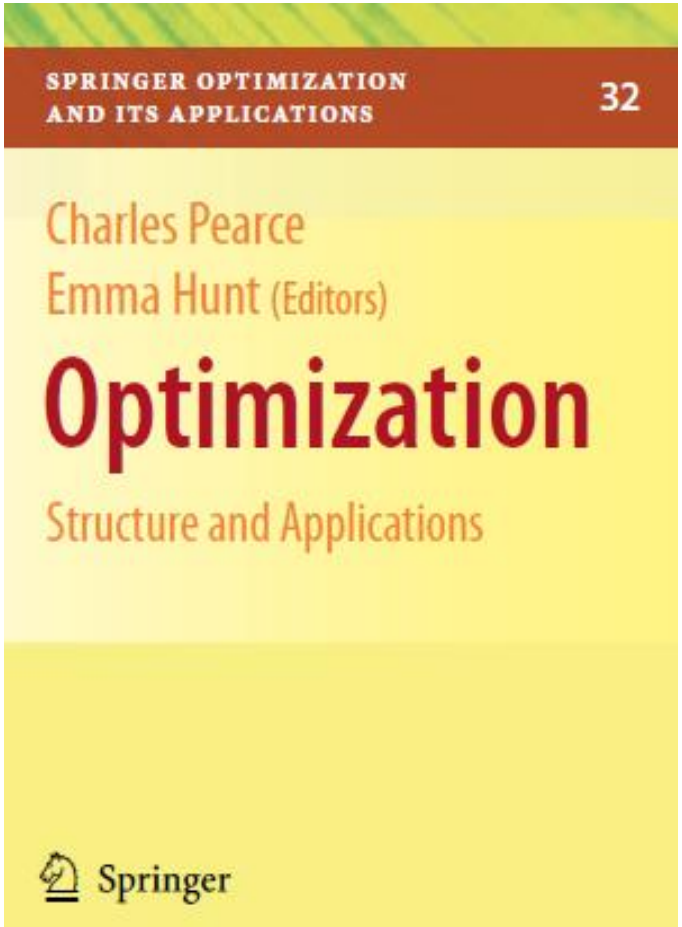
# Text Books


Optimization Techniques
and Applications
with Examples
XIN-SHE YANG
WILEY

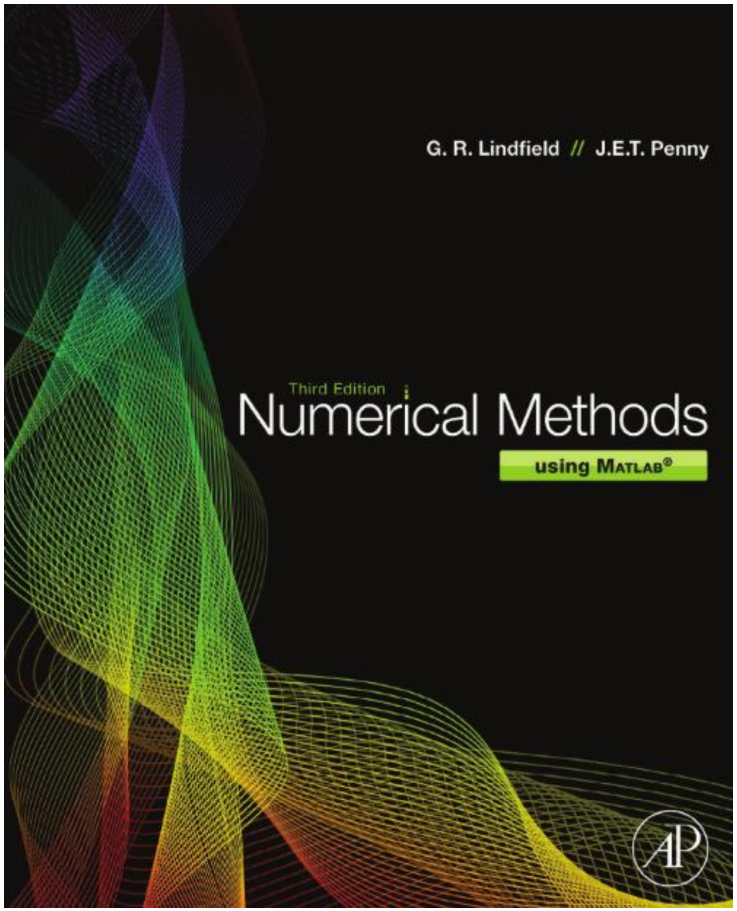

Energy Systems
Soliman Abdel-Hady Soliman
Abdel-Aal Hassan Mantawy
Modern Optimization Techniques with Applications in Electric Power Systems
Springer

# Text Books


ELSEVIER
ELSEVIER INSIGHTS
NATURE-INSPIRED
OPTIMIZATION
ALGORITHMS
XIN-SHE YANG


SPRINGER OPTIMIZATION
AND ITS APPLICATIONS
32
Charles Pearce
Emma Hunt (Editors)
Optimization
Structure and Applications
Springer

# Text Books




## Engineering Optimization

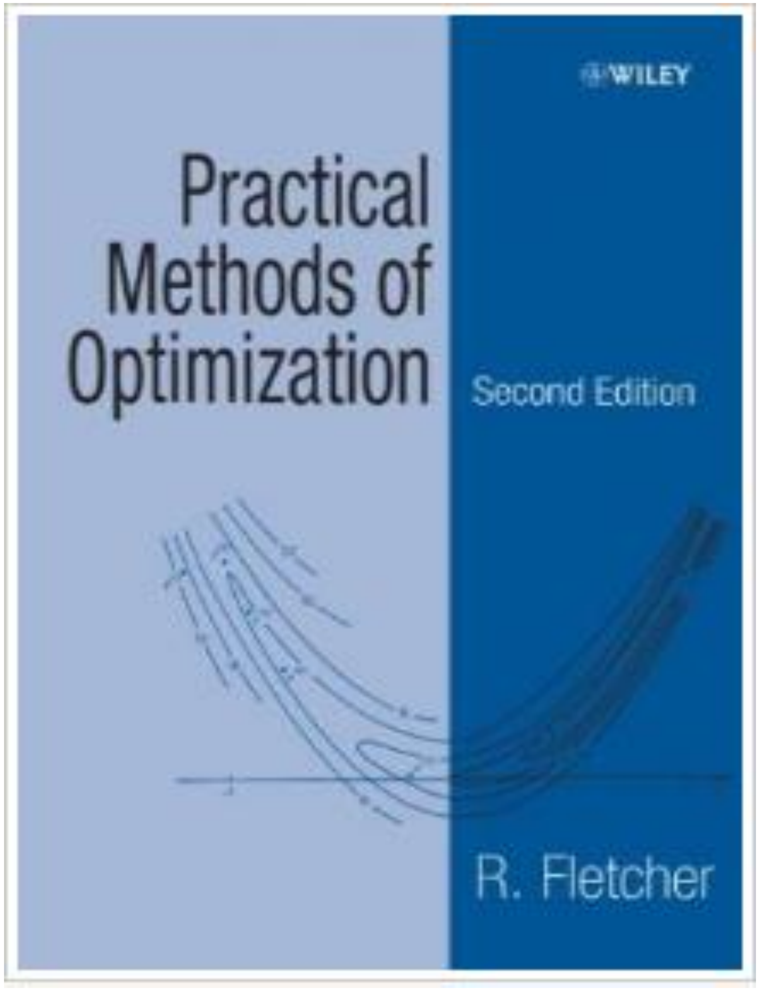
### An Introduction with Metaheuristic Applications

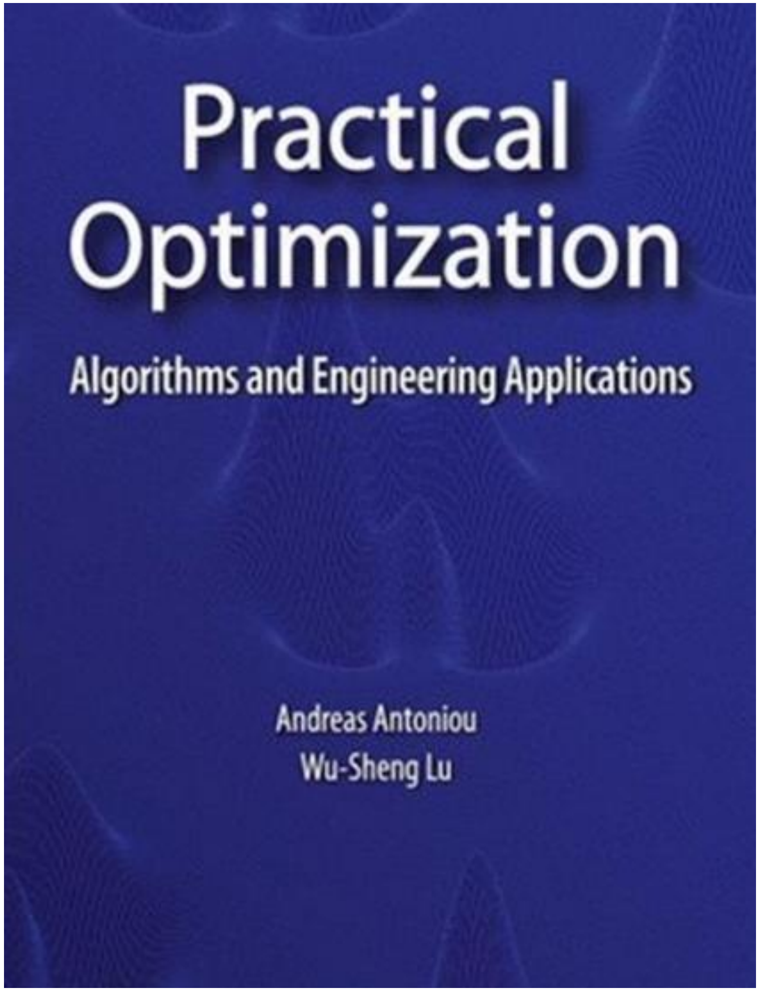
**Xin-She Yang**
*University of Cambridge*
*Department of Engineering*
*Cambridge, United Kingdom*



Prepared by: Diary R. Sulaiman

# Text Books


WILEY
Practical Methods of Optimization
Second Edition
R. Fletcher


Practical Optimization
Algorithms and Engineering Applications
Andreas Antoniou
Wu-Sheng Lu

Singiresu S. Rao

“Engineering Optimization: Theory and Practice.“

4th Edition, Wiley ISBN: 978-0-470-18352-6

Thomas Weise

“Global Optimization Algorithms– Theory and Application.”

Newest Version: http://www.it-weise.de/

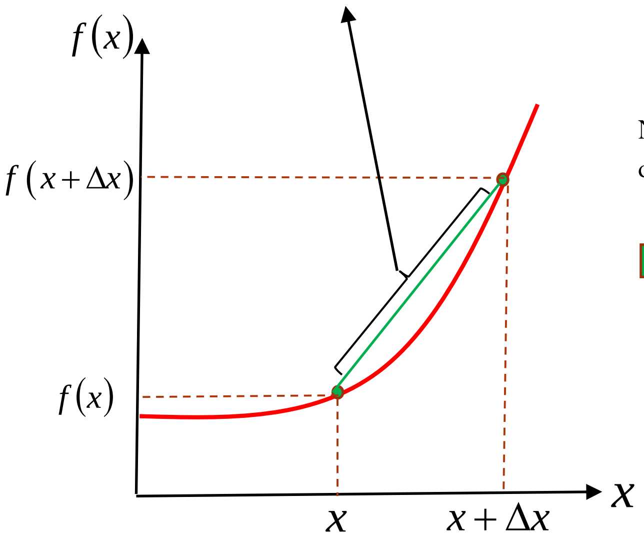
Bernhard Korte, Jens Vygen

“Combinatorial Optimization-Theory and Algorithms”

Second Edition

Imagine calculating the slope between two distinct points on a function

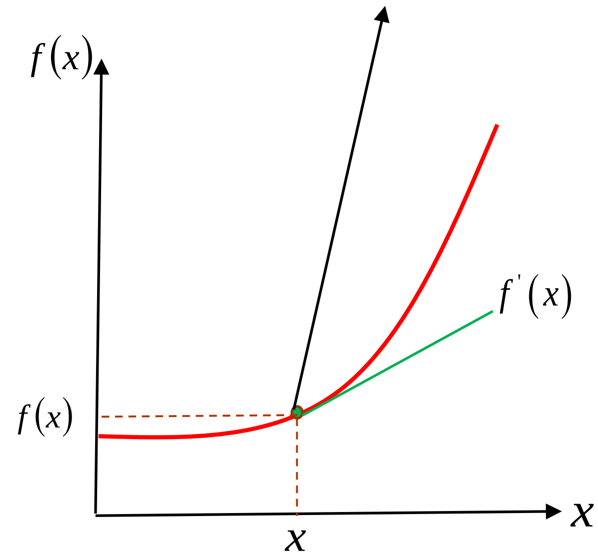
$$slope = \frac{f(x+\Delta x) - f(x)}{\Delta x}$$




Now, let those two points get closer and closer to each other

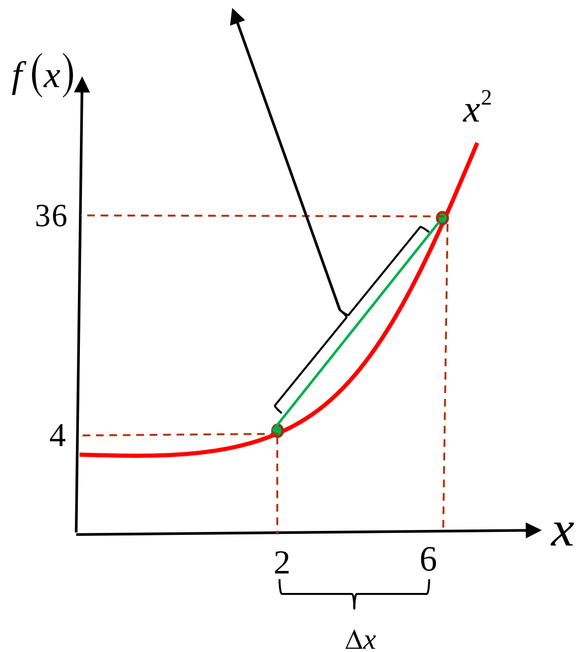
The derivative is the result of the distance between those two points approaching zero

$$slope = \lim_{\Delta x \to 0}\left[\frac{f(x+\Delta x) - f(x)}{\Delta x}\right] = f'(x)$$



# Let's try one numerically…

$$slope = \frac{36-4}{4} = 8$$



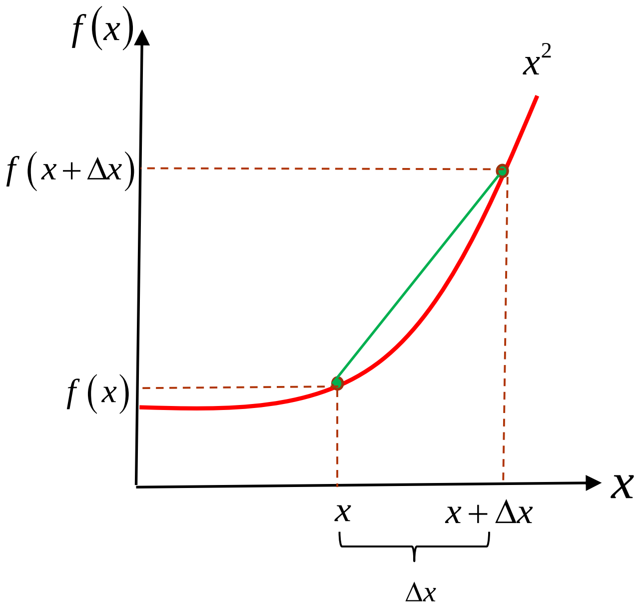

$$slope = \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

2

| $\Delta x$ | $f(x+\Delta x)$ | **Slope** |
|---|---|---|
| 4 | 36 | 8 |
| 2 | 16 | 6 |
| 1 | 9 | 5 |
| .5 | 6.25 | 4.5 |
| .25 | 5.0625 | 4.25 |
| .1 | 4.41 | 4.1 |
| .05 | 4.2025 | 4.05 |
| .01 | 4.0401 | 4.01 |
| .001 | 4.004001 | 4.001 |

The slope gets closer and closer to 4

# Or, in general…




$$slope = \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

⇓

$$slope = \frac{(x+\Delta x)^2 - x^2}{\Delta x}$$

⇓

$$slope = \frac{x^2 + 2x\Delta x - \Delta x^2 - x^2}{\Delta x}$$

⇓

$$slope = \frac{2x\Delta x - \Delta x^2}{\Delta x}$$

⇓

$$slope = 2x - \Delta x$$

⇒ Let the change in x go to 0

$$slope = 2x$$

$$f'(2) = 4$$

# Some useful derivatives

Linear Functions

$$f(x) = Ax \Rightarrow f'(x) = A$$

Example: $f(x) = 4x$

$$f'(x) = 4$$

Exponents

$$f(x) = Ax^n \Rightarrow f'(x) = nAx^{n-1}$$

Example: $f(x) = 3x^5$

$$f'(x) = 15x^4$$

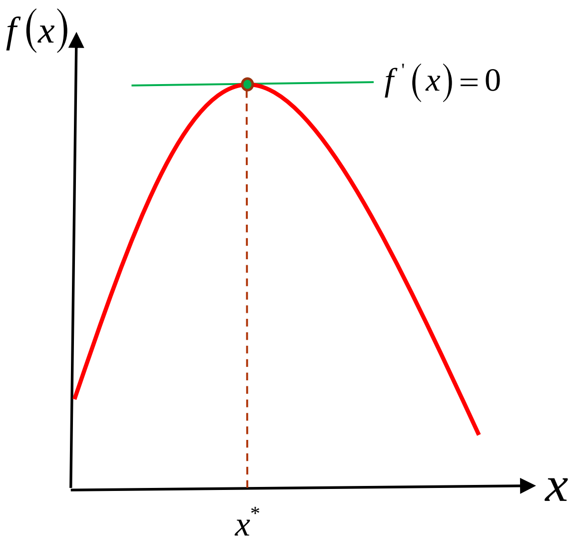
Logarithms

$$f(x) = A\ln(x) \Rightarrow f'(x) = \frac{A}{x}$$

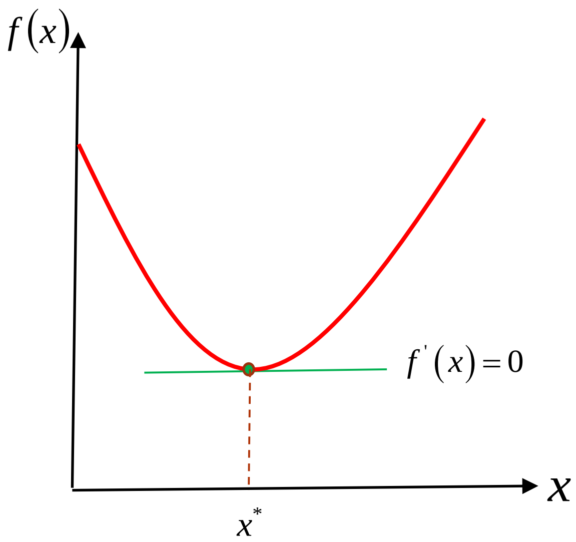
Example: $f(x) = 12\ln(x)$

$$f'(x) = \frac{12}{x}$$

A necessary condition for a maximum or a minimum is that the derivative equals zero

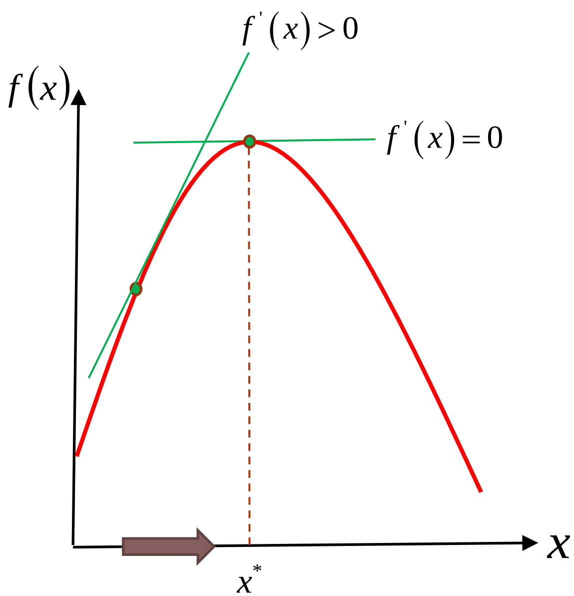



OR

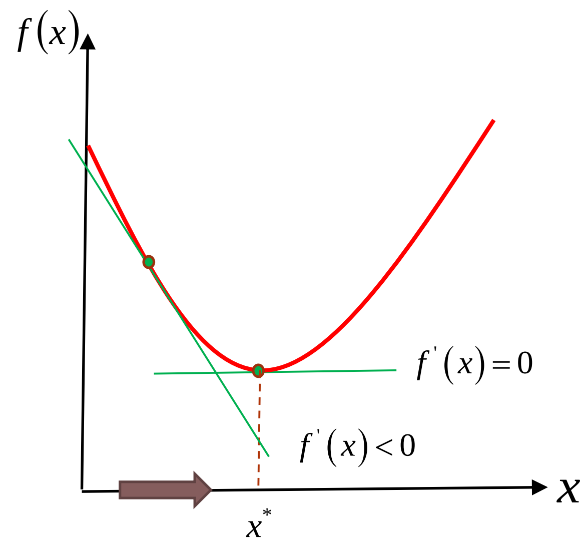



But how can we tell which is which?

While the first derivative measures the slope (change in the value of the function), the second derivative measures the change in the first derivative (change in the slope)




As x increases, the slope is decreasing } $f''(x) < 0$




As x increases, the slope is increasing } $f''(x) > 0$

Modern Optimization Techniques

MSc Course

Second Semester

2020-2021

# END of CH1

# Review and Introduction

Modern Computer Architecture

MSc Course

First Semester

# CH2
# Background


Prepared by: Diary R. Sulaiman

# Principles of Optimisation

***Typical engineering problem***: You have a ***process*** that can be represented by a ***mathematical model***. You also have a ***performance criterion*** such as minimum of ...

The goal of optimisation is to find the values of the ***variables*** in the process that yield the ***best value of the performance criterion.***

Two elements of an optimisation problem:

(i) *process or model*

(ii) *performance criterion*

## **Some typical performance criteria:**

- maximum profit
- minimum cost
- minimum effort
- minimum error
- minimum waste
- maximum throughput
- best product quality

Note the need to express the performance criterion in *mathematical form*.

*Static optimisation*:

The values of variables are requested that give minimum or maximum to an objective function (variables have numerical values)

$$\underset{\mathbf{x}}{\text{mininimize}}\, f(\mathbf{x})$$

*Dynamic optimisation*:

The variables as functions of time are requested that give minimum or maximum to an objective function (variables are functions of time)

$$\underset{\mathbf{y}(t)}{\text{mininimize}}\, J[\mathbf{y}(t)]$$

# Dependence on time

## Discrete

**Solution:**
**a sequence of optimal decisions in discrete time**

## Continuous

**Solution:**
**a time path or curve of optimal decisions in continuous time**

# Essential Features

Every optimisation problem contains three essential categories:

1. At least one ***objective function*** to be optimised

2. ***Equality constraints***

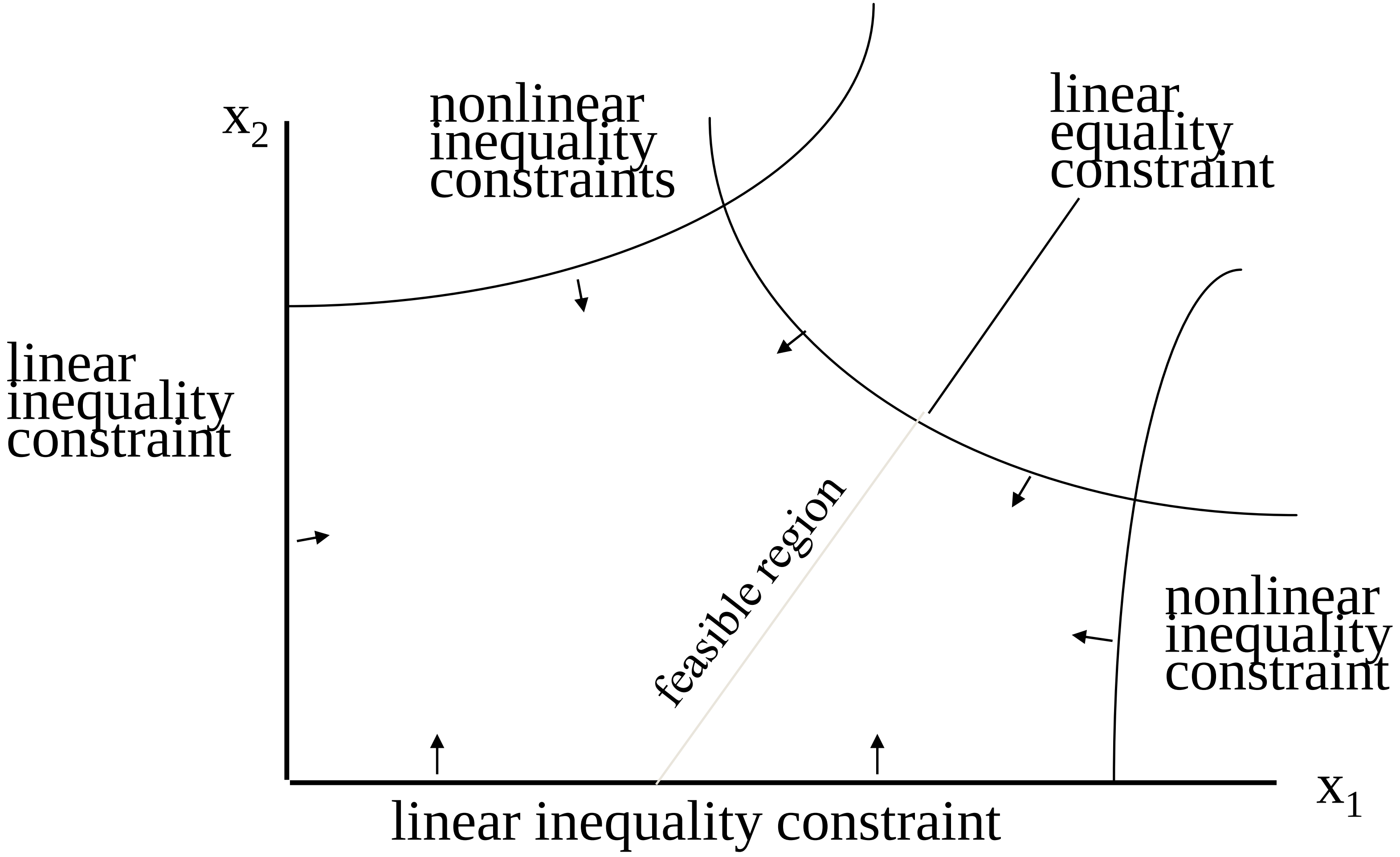
3. ***Inequality constraints***

# Objective Functions

Examples of Objective Functions :

- minimization of weight of the system,
- minimization of size of the system,
- maximization of efficiency,
- minimization of fuel consumption,
- minimization of exergy destruction,
- maximization of the net power density,
- minimization of emitted pollutants,
- minimization of life cycle cost (LCC) of the system,
- maximization of the internal rate of return (IRR),
- minimization of the payback period (PBP),
- etc.

By a ***feasible solution***: The region of feasible solutions is called the ***feasible region***.



An optimal solution is a ***set of values of the variables*** that are contained in the ***feasible region*** and also provide the ***best value of the objective function***.

For a meaningful optimisation problem the model needs to be ***underdetermined***.

# Mathematical Description

Minimize : $f(\mathbf{x})$ objective function

Subject to: $\begin{cases} \mathbf{h}(\mathbf{x}) = \mathbf{0} & \text{equality constraints} \\ \mathbf{g}(\mathbf{x}) \geq \mathbf{0} & \text{inequality constraints} \end{cases}$

where $\mathbf{x} \in \Re^n$, is a vector of n variables $(x_1, x_2, \cdots, x_n)$

$\mathbf{h}(\mathbf{x})$ is a vector of equalities of dimension $m_1$

$\mathbf{g}(\mathbf{x})$ is a vector of inequalities of dimension $m_2$

# Steps Used To Solve Optimisation Problems

1. Analyse the process in order to make a list of all the variables.
2. Determine the optimisation criterion and specify the objective function.
3. Develop the mathematical model of the process to define the equality and inequality constraints. Identify the independent and dependent variables to obtain the number of degrees of freedom.
4. If the problem formulation is too large or complex simplify it if possible.
5. Apply a suitable optimisation technique.
6. Check the result and examine it's ***sensitivity*** to changes in model parameters and assumptions.

# Classification of Optimisation Problems

## Properties of $f(\mathbf{x})$

- single variable or multivariable
- linear or nonlinear
- sum of squares
- quadratic
- smooth or non-smooth
- sparsity???

## Properties of h(x) and g(x)

- simple bounds
- smooth or non-smooth
- sparsity
- linear or nonlinear
- no constraints

# Properties of variables x

- time variant or invariant
- continuous or discrete
- take only integer values
- mixed

# Obstacles and Difficulties

- Objective function and/or the constraint functions may have finite ***discontinuities*** in the continuous parameter values.

- Objective function and/or the constraint functions may be ***non-linear functions*** of the variables.

- Objective function and/or the constraint functions may be defined in terms of ***complicated interactions*** of the variables. This may prevent calculation of ***unique values*** of the variables at the optimum.

- Objective function and/or the constraint functions may ***exhibit nearly "flat" behaviour*** for some ranges of variables or ***exponential behaviour*** for other ranges. This causes the problem to be ***insensitive, or too sensitive***.

- The problem may exhibit ***many local optima*** whereas the ***global optimum is sought***. A solution may ***less satisfactory than another solution elsewhere*** be obtained that is.

- ***Absence of a feasible region.***

- ***Model-reality differences.***

# Typical Examples of Application static optimisation

- Plant design (sizing and layout).
- Operation (best steady-state operating condition).
- Parameter estimation (model fitting).
- Allocation of resources.
- Choice of controller parameters (e.g. gains, time constants) to minimise a given performance index (e.g. overshoot, settling time, integral of error squared).

# Dynamic optimisation

- Determination of a control signal $u(t)$ to transfer a dynamic system from an initial state to a desired final state to satisfy a given performance index.
- Optimal plant start-up and/or shut down.
- Minimum time problems

# basic principles of static optimisation theory

## Continuity of Functions

Functions containing discontinuities can cause difficulty in solving optimisation problems.

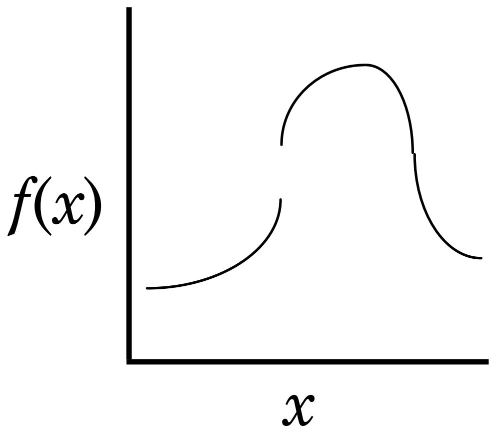
*Definition*: A function of a single variable $x$ is continuous at a point $x_o$ if:

(a) $f(x_o)$ exists

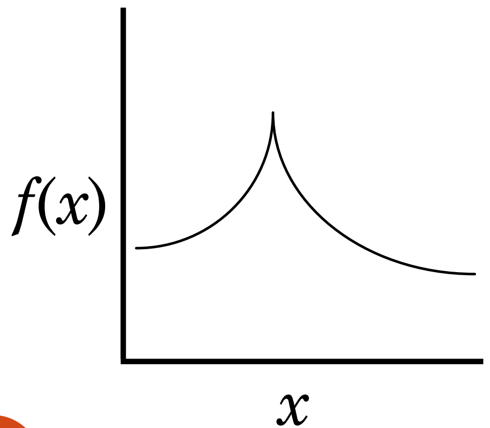
(b) $\lim_{x \to x_o} f(x)$ exists

(c) $\lim_{x \to x_o} f(x) = f(x_o)$

If $f(x)$ is continuous at every point in a region $R$, then $f(x)$ is said to be continuous throughout $R$.




$f(x)$ is discontinuous.




$f(x)$ is continuous, but

$f'(x) \equiv \frac{df}{dx}(x)$ is not.

# Unimodal and Multimodal Functions

A ***unimodal function*** $f(x)$ (in the range specified for $x$) has a single extremum (minimum or maximum).

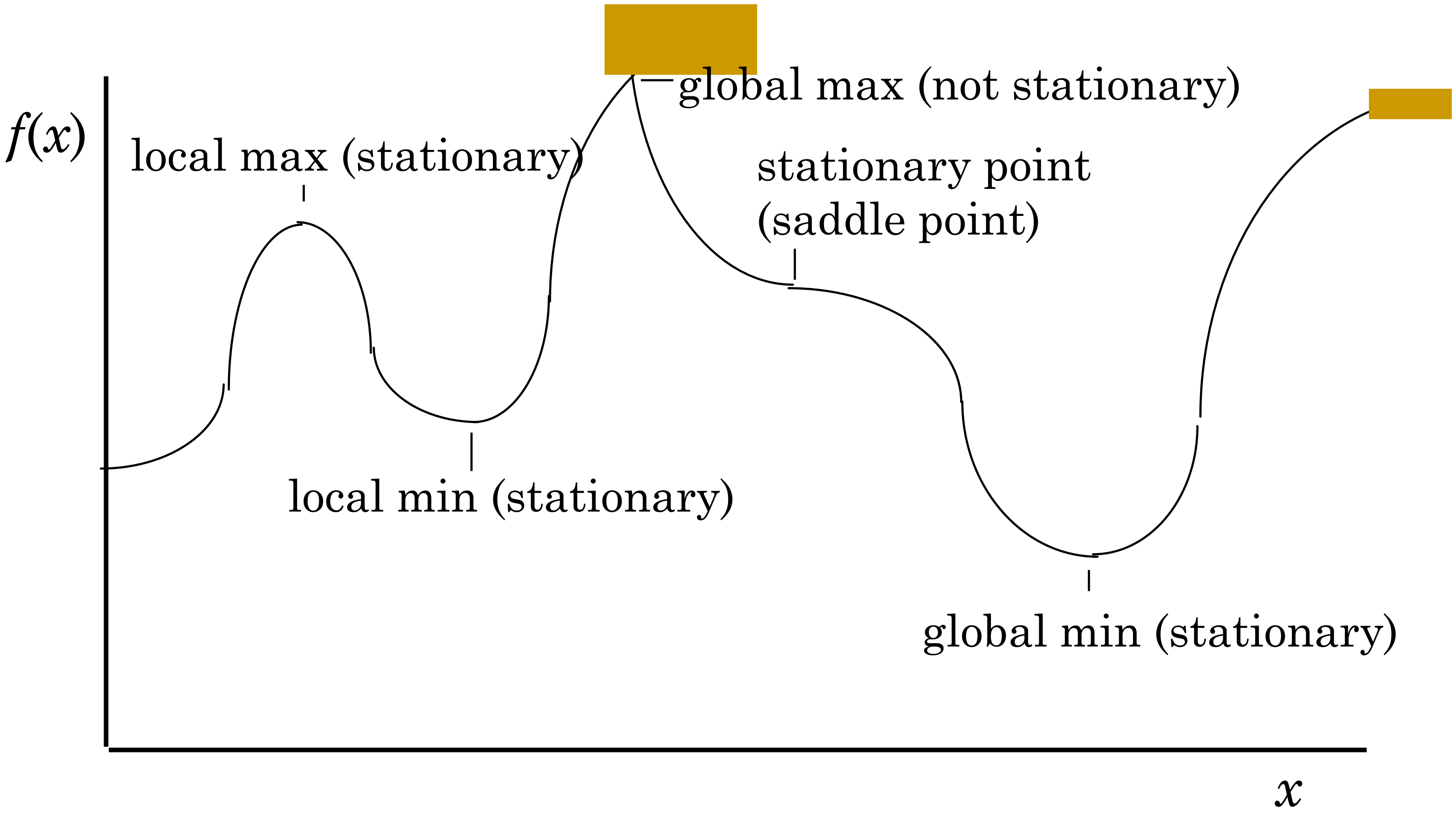
A ***multimodal function*** $f(x)$ has two or more extrema.

If $f'(x) = 0$ at the extremum, the point is called a ***stationary point***

There is a distinction between the ***global extremum*** (the biggest or smallest between a set of extrema) and ***local extrema*** (any extremum).

***Note***: many numerical procedures terminate at a local extremum.

# A multimodal function



# Multivariate Functions - Surface and Contour Plots

We shall be concerned with basic properties of a scalar function $f(\mathbf{x})$ of $n$ variables $(x_1,\ldots,x_n)$.

If $n = 1$, $f(x)$ is a ***univariate function***
If $n > 1$, $f(\mathbf{x})$ is a ***multivariate function.***

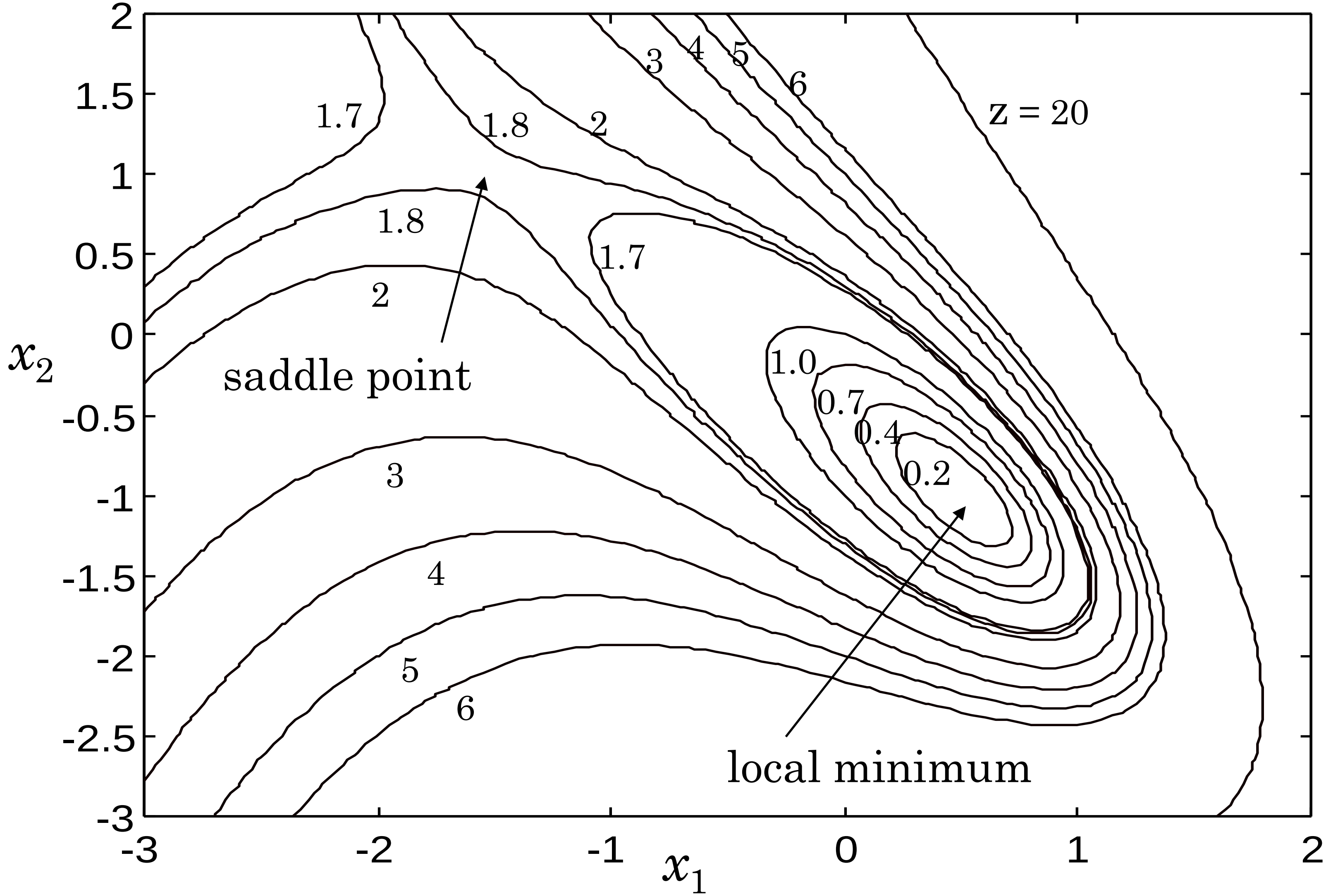
For any multivariate function, the equation $z = f(\mathbf{x})$ defines a ***surface*** in $n$+1 dimensional space $\Re^{n+1}$

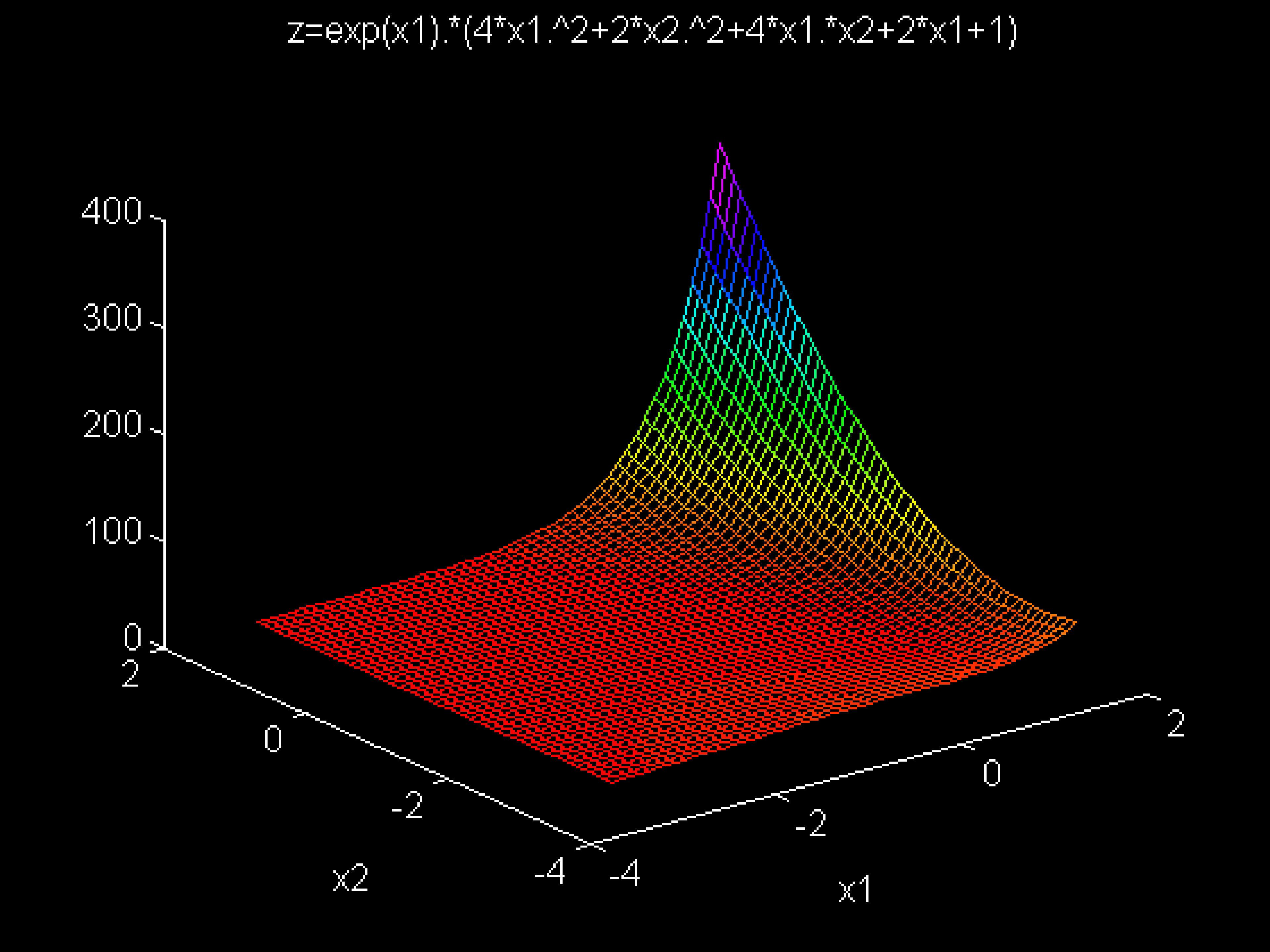
In the case $n = 2$, the points $z = f(x_1,x_2)$ represent a three dimensional surface.

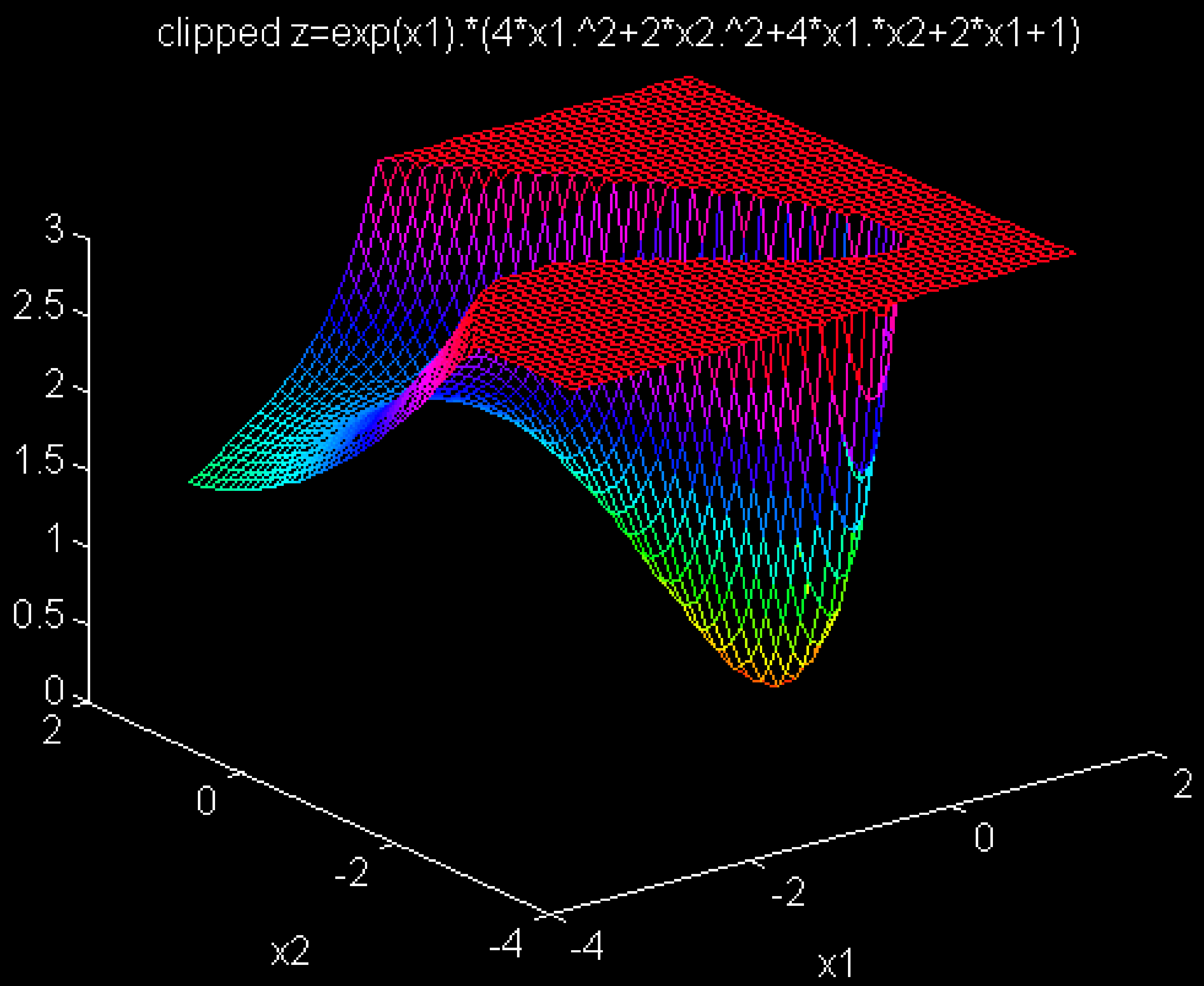
Let $c$ be a particular value of $f(x_1,x_2)$. Then $f(x_1,x_2) = c$ defines a curve in $x_1$ and $x_2$ on the plane $z = c$.

If we consider a selection of different values of $c$, we obtain a family of curves which provide a ***contour map*** of the function $z = f(x_1,x_2)$.

# contour map of $z = e^{x_1}(4x_1^2 + 2x_2^2 + 4x_1x_2 + 2x_2 + 1)$




z=exp(x1).*(4*x1.^2+2*x2.^2+4*x1.*x2+2*x1+1)
400
300
200
100
0
2
0
-2
-4
x2
-4
-2
0
2
x1


clipped z=exp(x1).*(4*x1.^2+2*x2.^2+4*x1.*x2+2*x1+1)
3
2.5
2
1.5
1
0.5
0
2
0
-2
-4
x2
-4
-2
0
2
x1

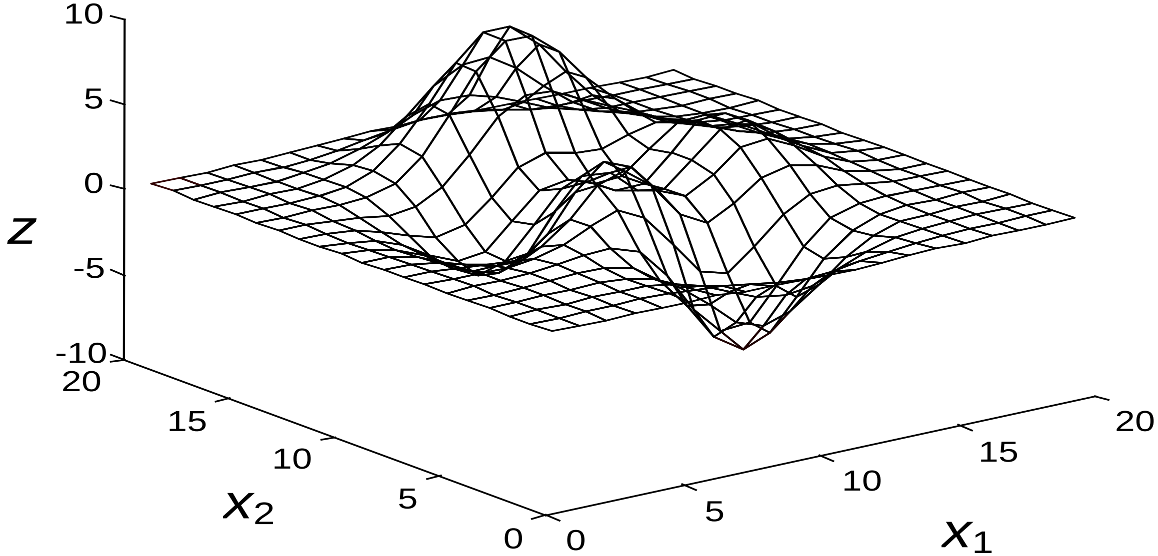
# Example: Surface and Contour Plots of "Peaks" Function

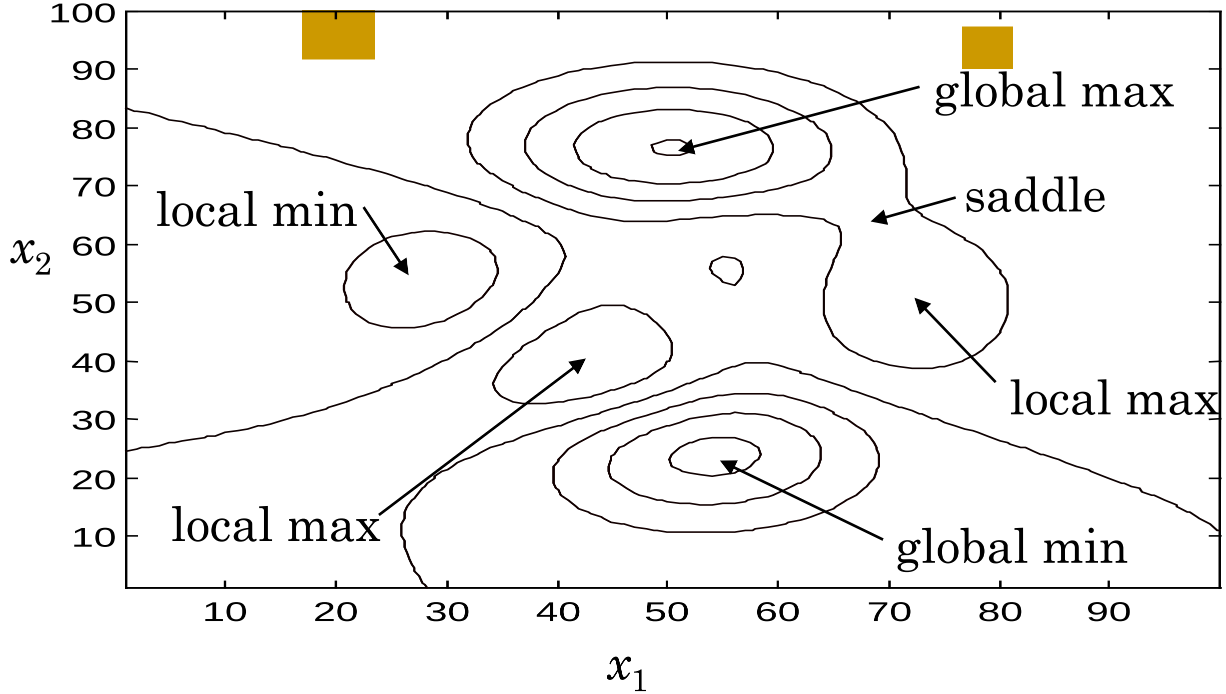
$$z = 3(1 - x_1)^2 \exp\left(-x_1^2 - (x_2 + 1)^2\right)$$

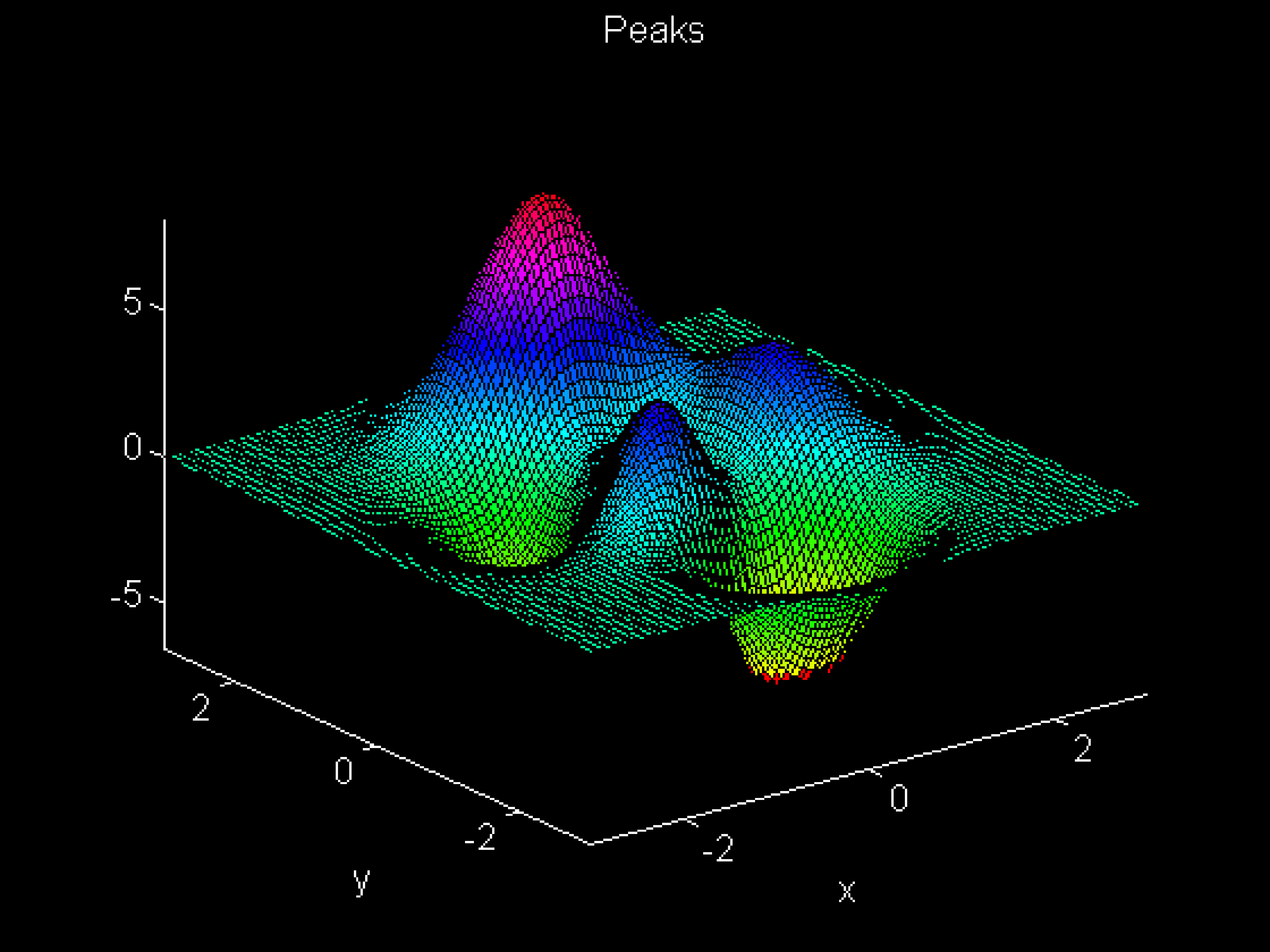
$$-10(0.2x_1 - x_1^3 - x_2^5)\exp(-x_1^2 - x_2^2)$$

$$-1/3 \exp\left(-(x_1 + 1)^2 - x_2^2\right)$$

# multimodal!


10
5
0
-5
-10
z
20
15
10
5
0
x2
0
5
10
15
20
x1


100
90
80
70
60
50
40
30
20
10
x2
10
20
30
40
50
60
70
80
90
x1
global max
saddle
local min
local max
local max
global min


Peaks
5
0
-5
2
0
-2
y
-2
0
2
x

Modern Optimization Techniques

MSc Course

First Semester

2020-2021

# END of CH2

# CH1-CH2

# Review and Introduction



Prepared by: Diary R. Sulaiman