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

First Semester

CH2
Background

Prepared by: Diary R. Sulaiman

Start with a Dream…

- Suppose you have a problem
- You don't know how to solve it
- What can you do?
- Can you use a computer to somehow find a solution for you?
- This would be nice! Can it be done?

A dumb solution

- A "*blind generate and test*" algorithm:
- Repeat
- Generate a random possible solution
- Test the solution and see how good it is
- Until solution is good enough

Can we use this dumb idea?

- Sometimes - ***YES***:
  - if there are only a few possible solutions
  - and you have enough time
  - then such a method *could be* used
- For most problems - ***NO***:
  - many possible solutions
  - with no time to try them all
  - so this method *can not* be used

A "less-dumb" idea

Generate

- ***Generate*** a ***set*** of random solutions

Repeat

Test

- ***Test*** each solution (rank them)

Remove some bad solutions from set

Duplicate some good solutions

Make small changes to some of them

Until best solution is good enough

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***

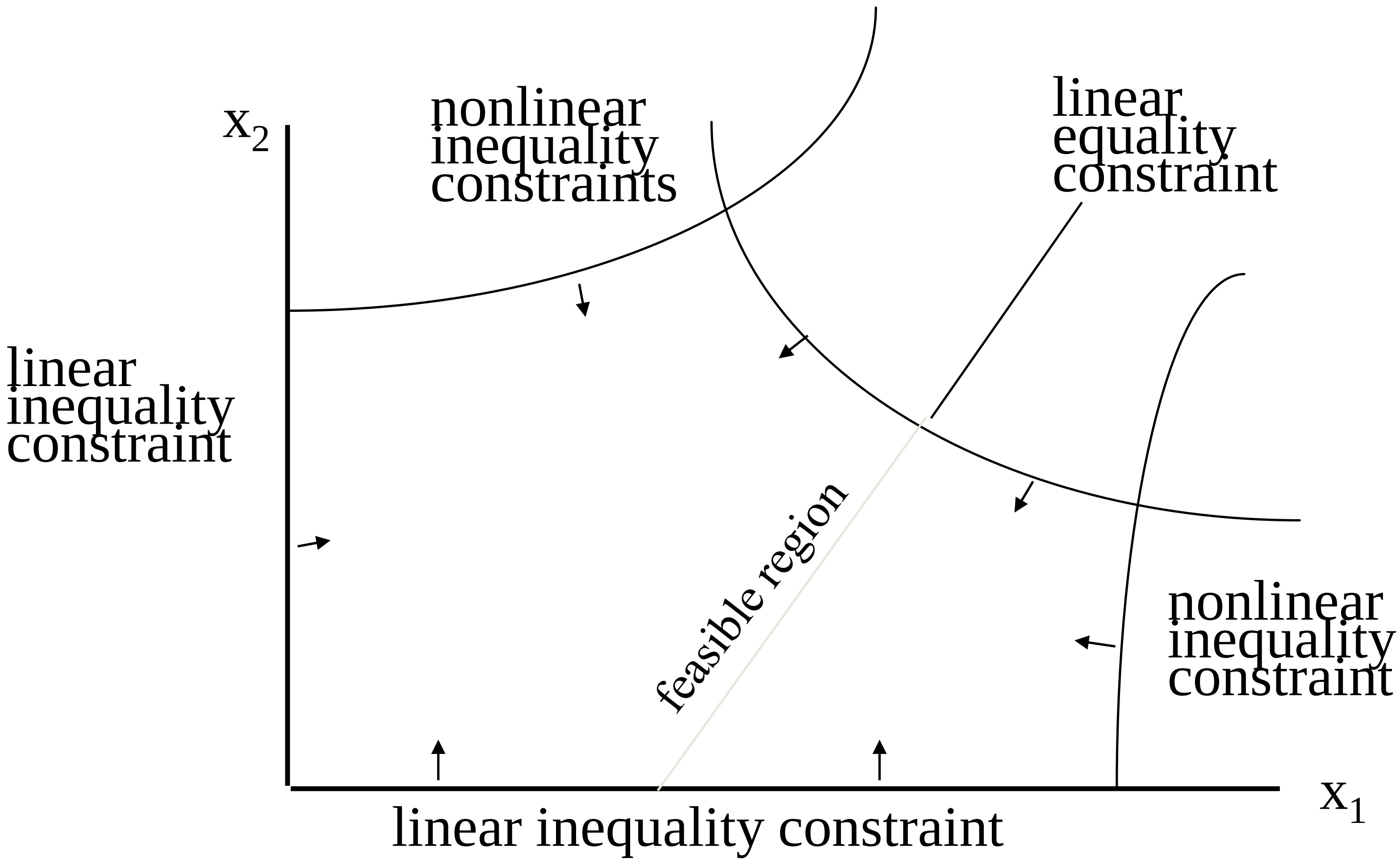
3. ***Inequality constraints***

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

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

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

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.

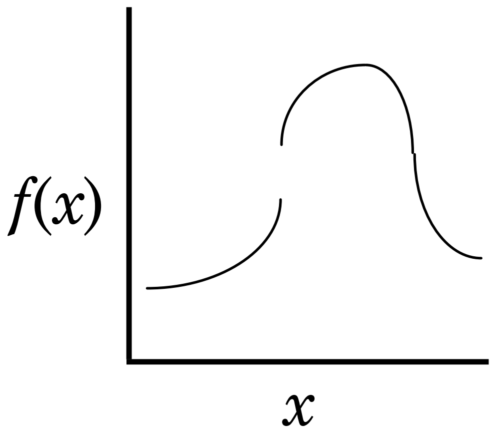
Definition: A function of a single variable x is continuous at a point x_o if:

$$\text{(a)}\quad f(x_o) \quad \text{exists}$$

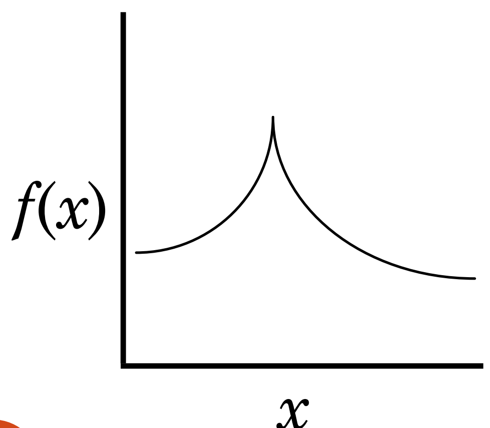
$$\text{(b)}\quad \lim_{x \to x_o} f(x) \quad \text{exists}$$

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

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).

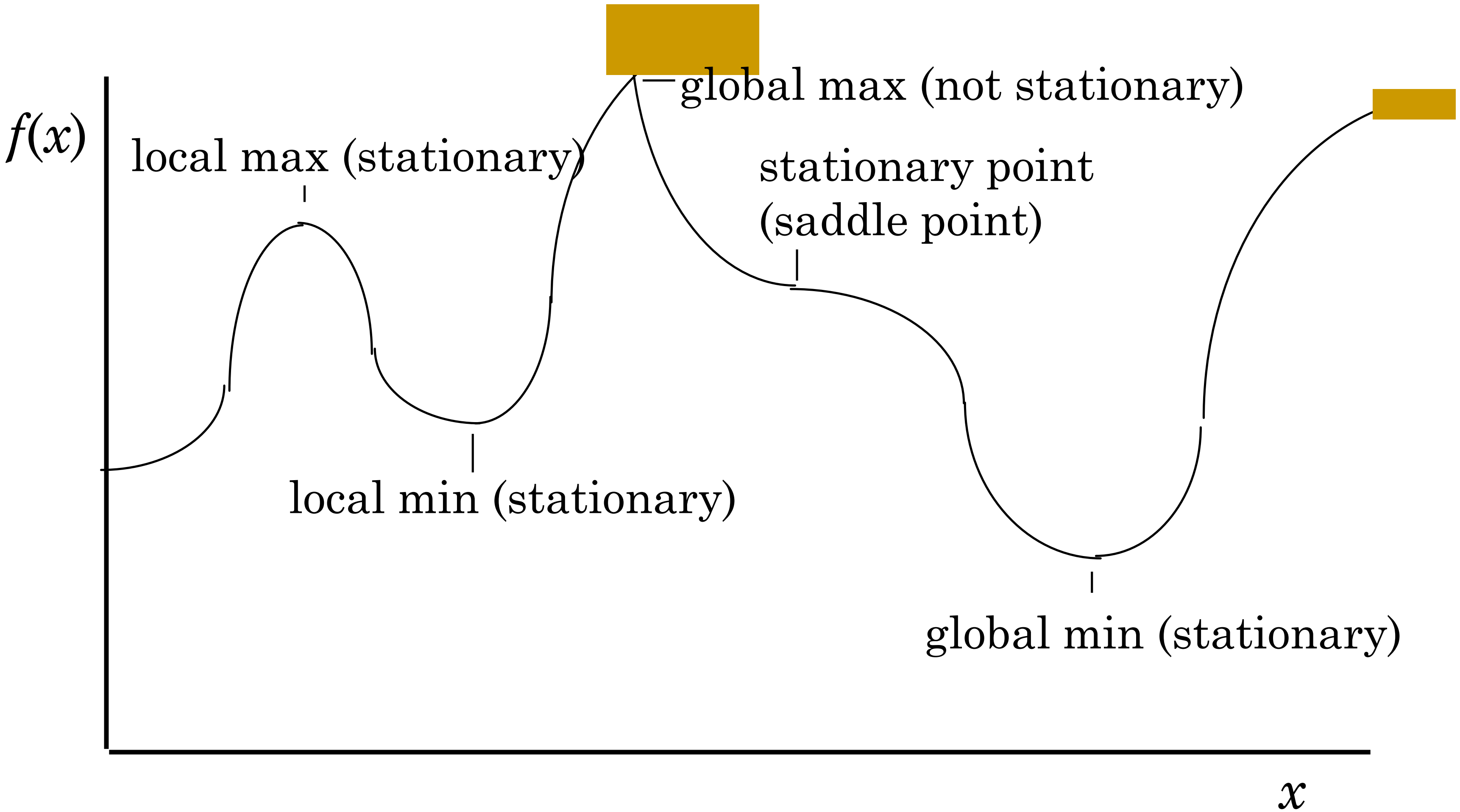
A ***multimodal function*** $f(x)$ has two or more extrema.

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.***

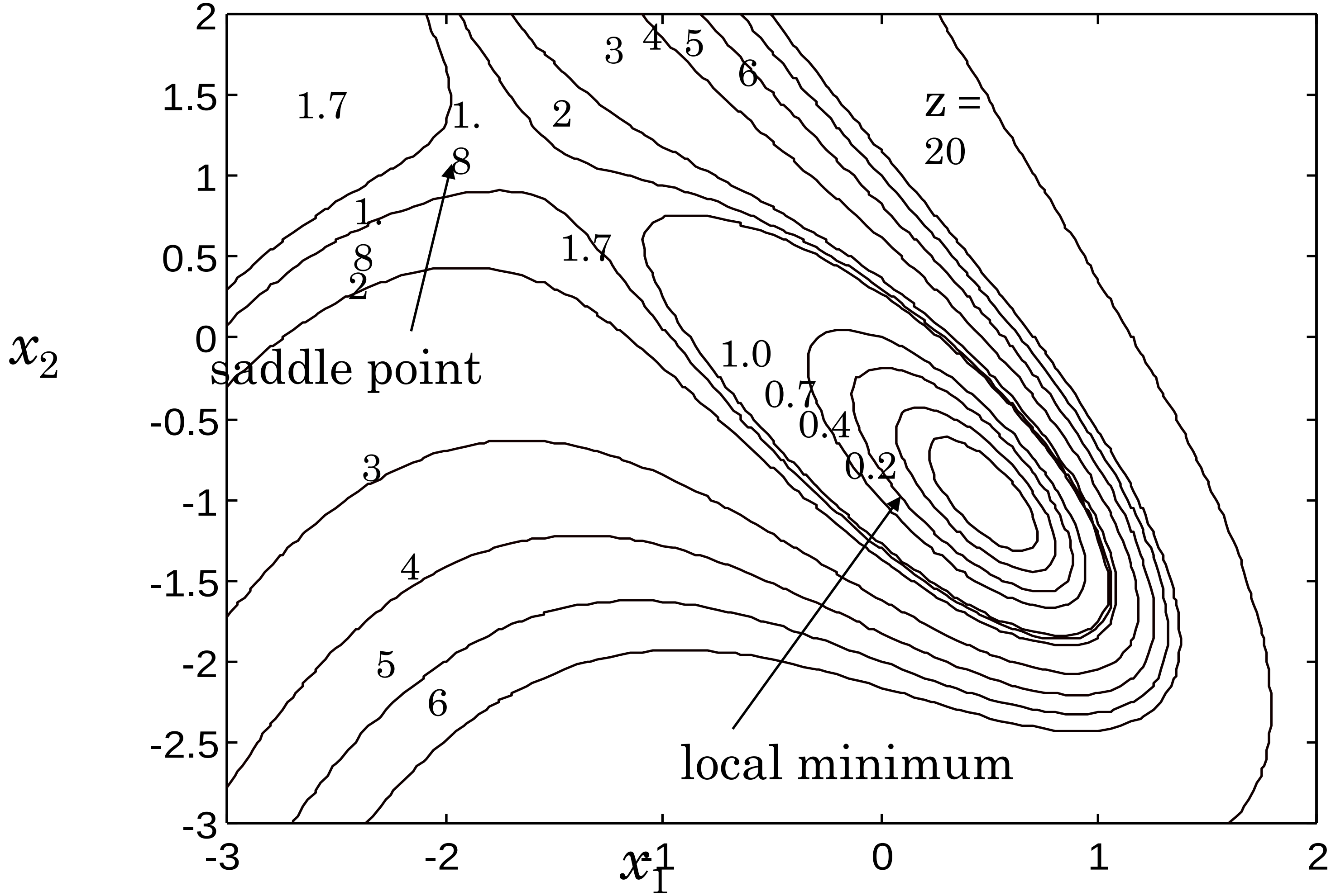
For any ***multivariate*** function, the equation $z = f(\mathbf{x})$ defines a ***surface*** in **n+1** dimensional space $\Re^{n+1}$

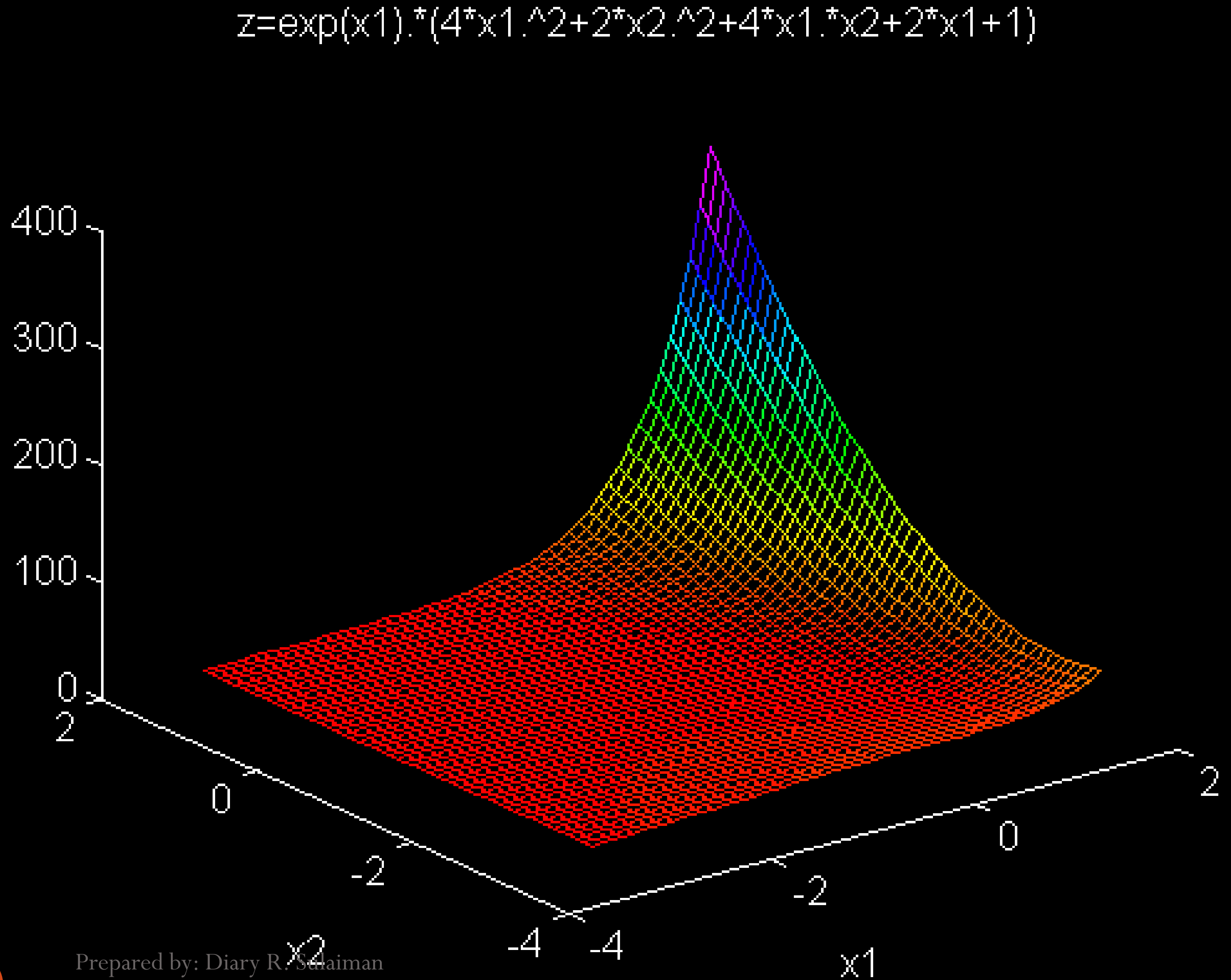
In the case $n = 2$, the points $z = f(x_1,x_2)$ represent a three dimensional surface.

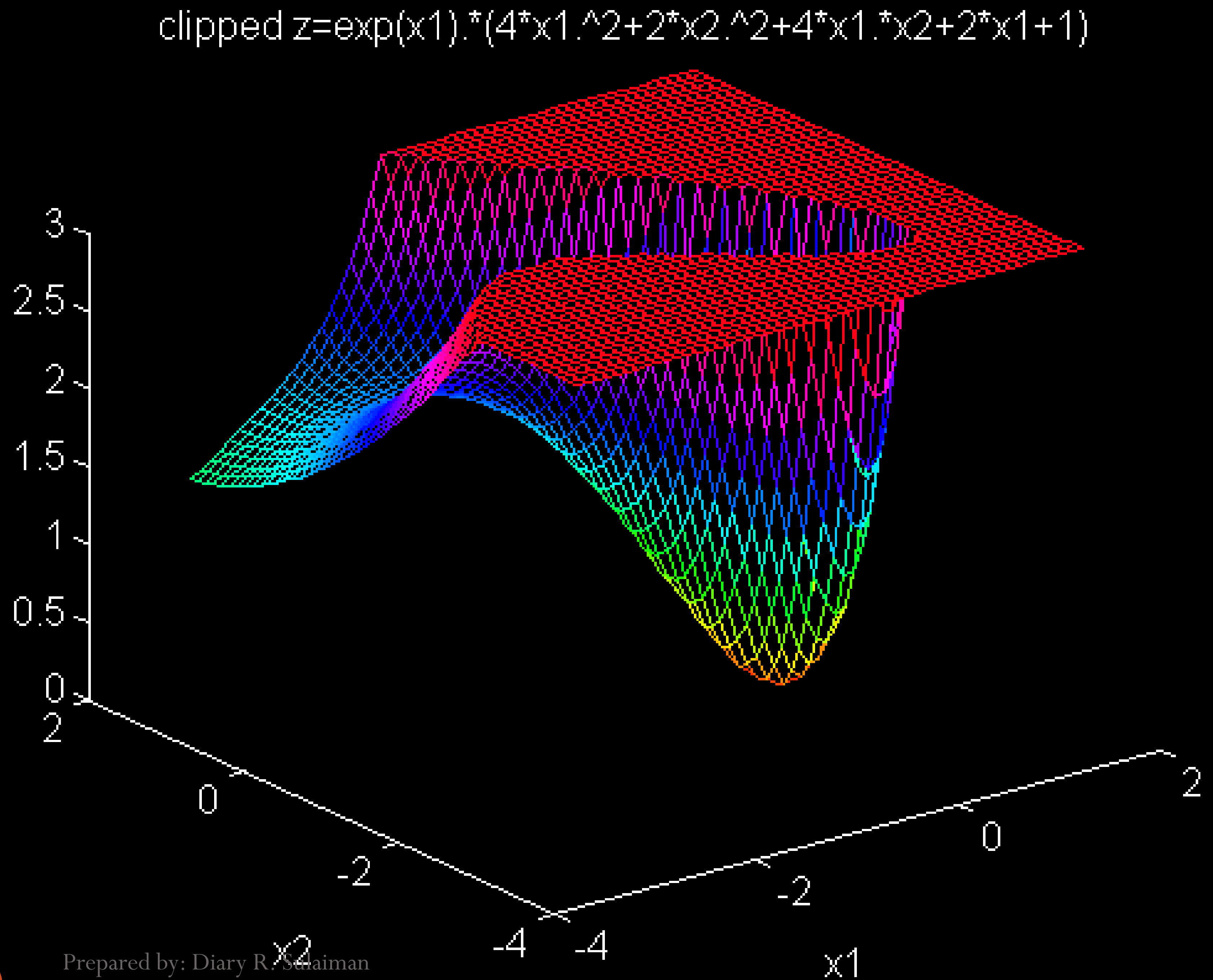
Let c be a particular value of $f(x_1,x_2)$. Then $f(x_1,x_2) = c$ defines a curve in x_1 and x_2 on the plane $z = c$.

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

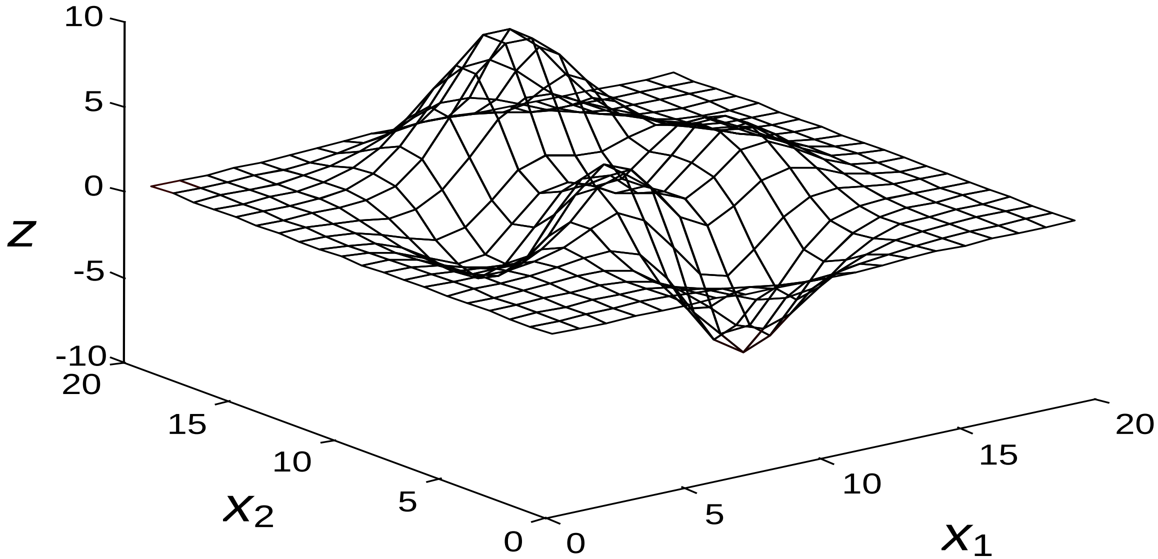
Example: Surface and Contour Plots of "Peaks" Function

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

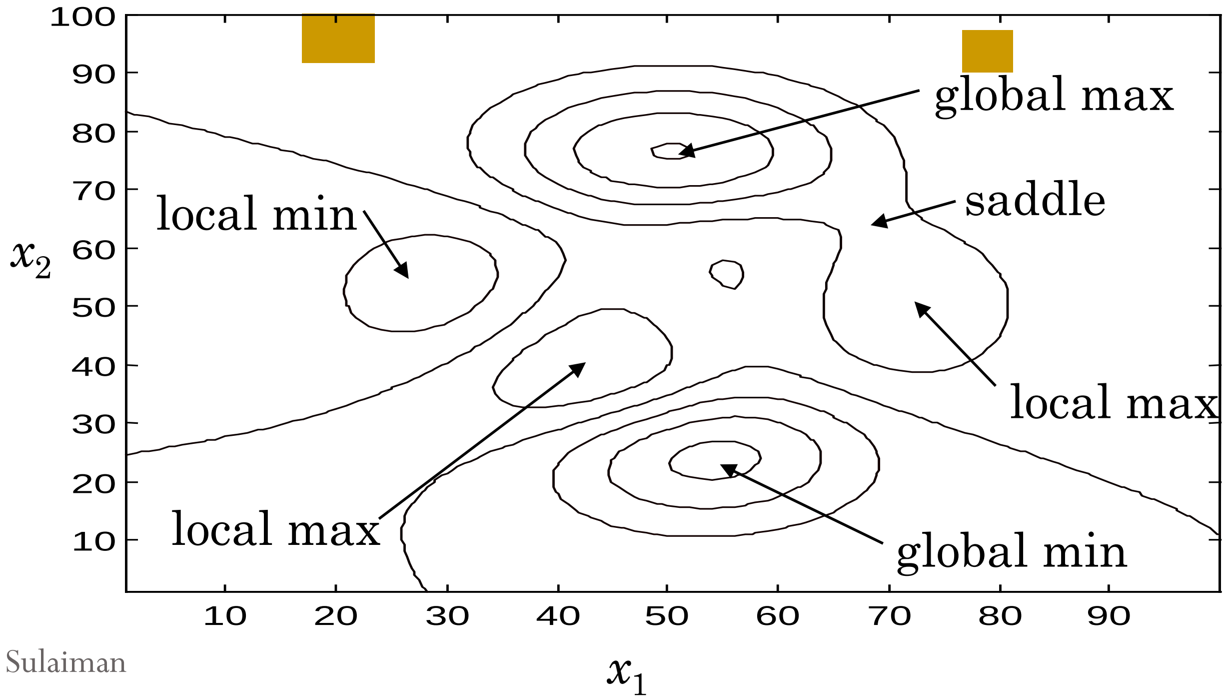
$$-10(0.2x_1 - x_1^3 - x_2^5)\exp(-x_1^2 - x_2^2)$$

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

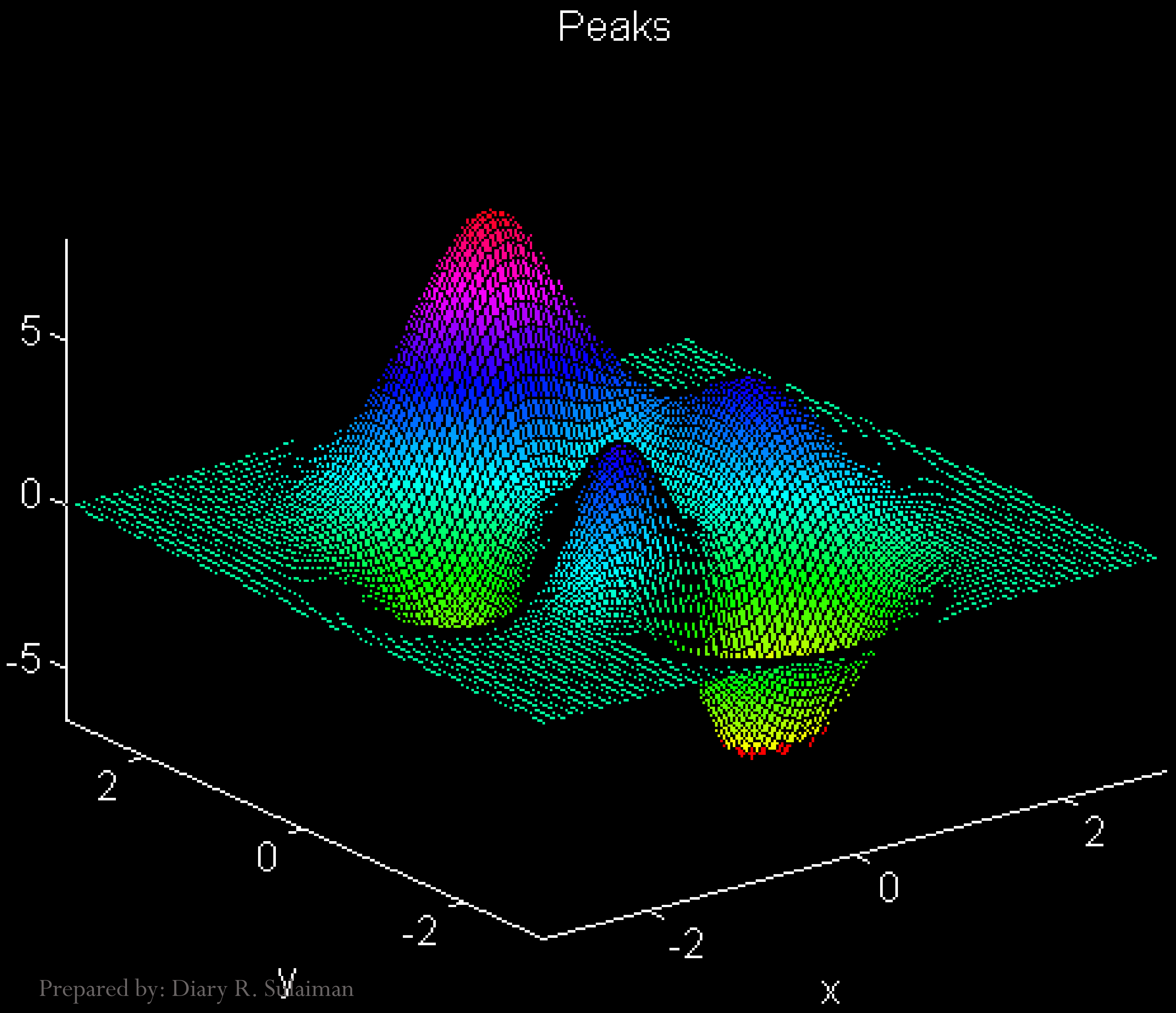
multimodal!



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



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


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

Modern Optimization Techniques

MSc Course

First Semester

2020-2021

END of CH2

Background

Prepared by: Diary R. Sulaiman