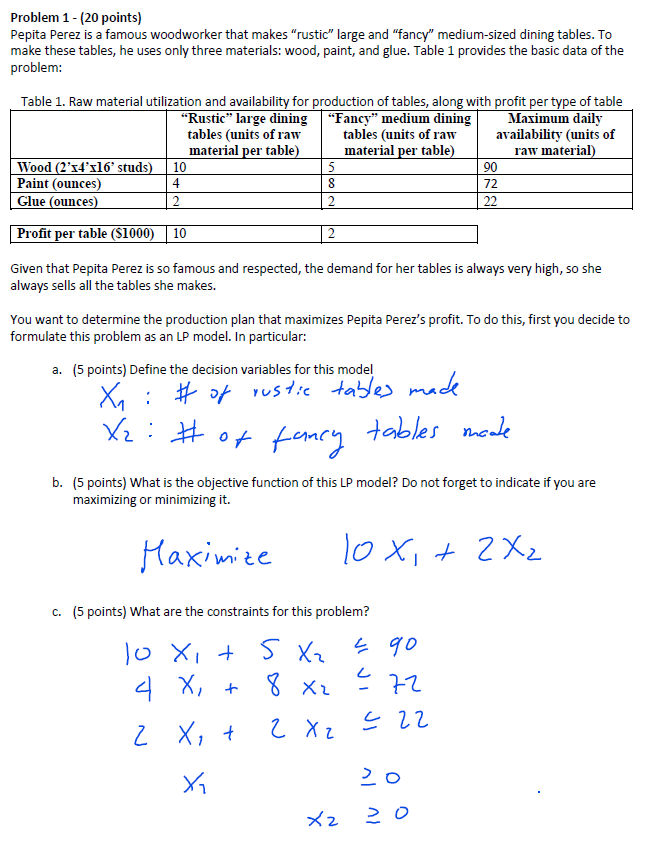
# Notes for Exam 1 – Systems Optimization

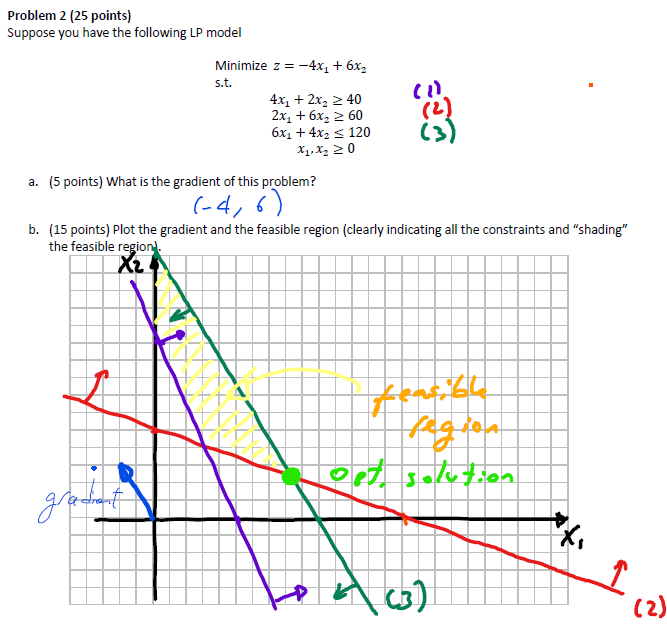
## Tips

* Inequality signs (<=, >=, etc… for graphs and systems)
* Be sure to check min/max before optimizing

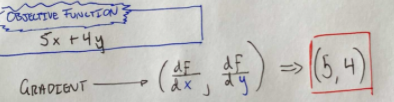
## Basic Setup to Linear Programming

****

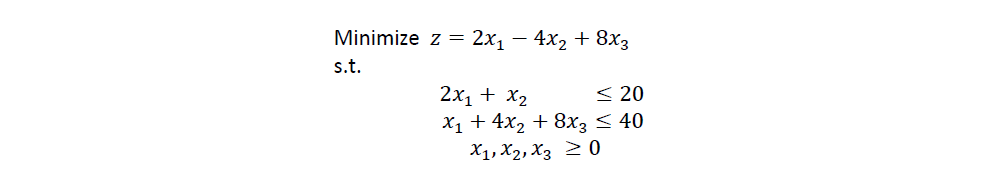
## Graph Optimization Problem

****

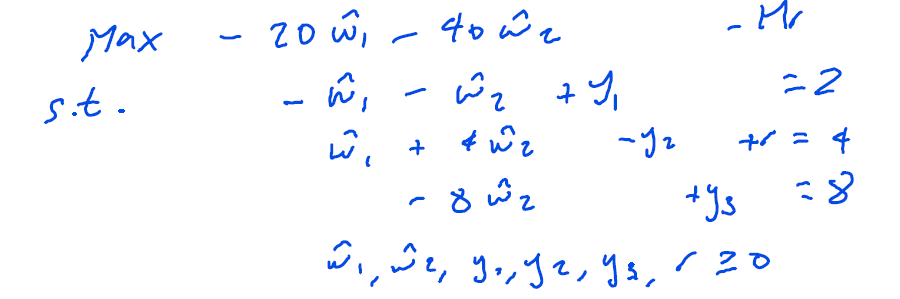
## Solving Gradient given objective function



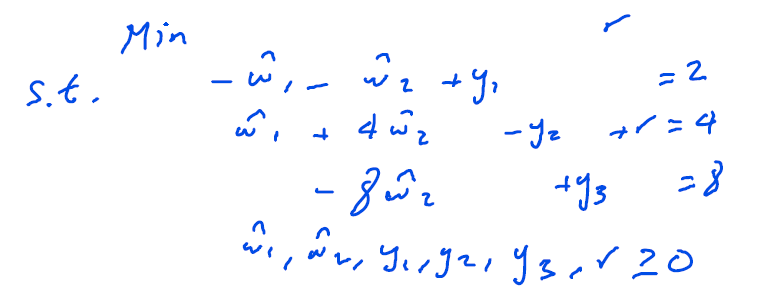
## Initialize Two-phase and Big-M – *Linear System*

****

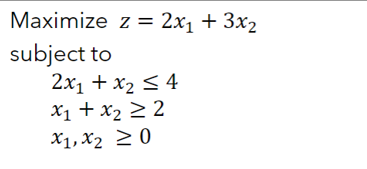
**Big M - Linear System**

****

**Two-phase – Linear System**

****

## Initialize Two-phase and Big-M – *Matrix*

****

**Big M – Matrix**

* -1 = slack var
* 1000 = slack row at *r*
* -1 = Z row at *r*
* RHS = positive
* LHS = same as original constraint (except slack var)
* Make r column identity, then optimize via simplex

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Basis** | **Z** | **x1** | **x2** | **s1** | **s2** | **r** | **sol** |
| Z | 1 | -2 | -3 | 0 | 0 | 1000 | 0 |
|  | 0 | 2 | 1 | 1 | 0 | 0 | 4 |
|  | 0 | 1 | 1 | 0 | -1 | 1 | 2 |

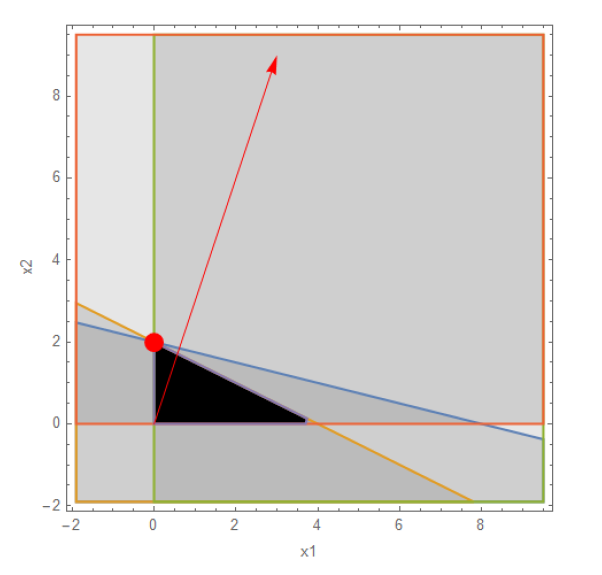
**Two Phase – Matrix**

* -1 = slack var
* 1 = slack row at *r*
* -1 = Z row at *r*
* RHS = positive
* LHS = same as original constraint (except slack var)
* Make r column identity, then optimize via simplex
* Drop R row once found initial basic solution

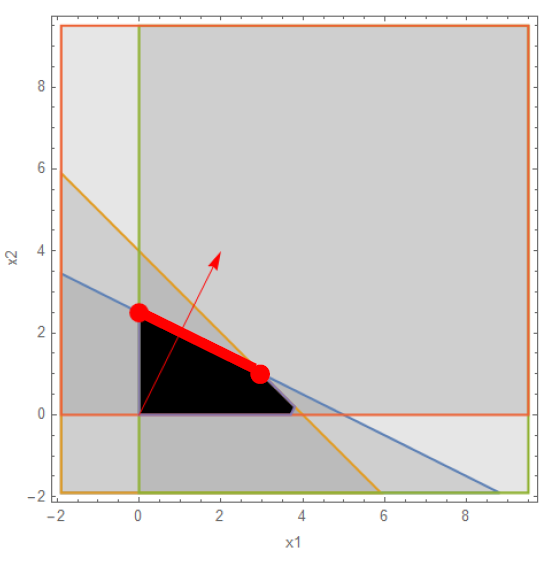
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Basis** | **z** | **x1** | **x2** | **s1** | **s2** | **r** | **sol** |
| z | 1 | -2 | -3 | 0 | 0 | -1 | 0 |
|  | 0 | 2 | 1 | 1 | 0 | 0 | 4 |
|  | 0 | 1 | 1 | 0 | -1 | 1 | 2 |

## Degeneracy and Alternative Optima

**Degeneracy:** Two constraints overlap at the optimal solution. Remove one constraint if this happens.

****

**Alternative Optima:** two feasible optimal solutions. Use path of least iterations.

****

## Python Code for Optimization - Gurobi

****