University of Southern California

ISE 536 - Linear Programming and Extensions

**Minimizing Production Cost of Material Handling Products**

**Team 4:**

Daniel Ley

Ruoqian Lan

Vaishnavi Guntupalli

Shuochen Tao

Date: December 8, 2021

Instructor: Sima Parisay, PhD

Teaching Assistant: Siyuan Yao

**Table of Contents**

[**Project Overview**](#_mxkamymh7mez) **5**

[**Linear Programming (LP) Model**](#_wj3qqphddtn8) **7**

[1.0 Introduction](#_xt9ffvabickz) 7

[2.0 Summary Table of Input Information](#_u5w08cfatb9g) 8

[2.1 Input Information for Rod Type A](#_sg1xrtiw4rfy) 8

[2.2 Input Information for Rod Type B](#_gui128yyoq3b) 9

[3.0 Summary Table of Output Information](#_8236gz6pqnxh) 10

[3.1 Output Analysis of Rod Type A](#_e72lrhnivwvh) 10

[3.2 Output Analysis of Rod Type B](#_xhvbv4qf3b3v) 11

[4.0 Linear Programming Model Summary](#_ki6blg6c77xy) 12

[5.0 Linear Programming Sensitivity Analysis](#_wwk07baiimz4) 12

[5.1 Rod B Sensitivity Analysis of Variables](#_ox0fqt8lg55y) 13

[5.2 Rod B Sensitivity Analysis of Constraints](#_vrworxlmfw9s) 14

[**Integer Linear Programming (ILP) Model**](#_eviint9i57cm) **16**

[1.0 Introduction](#_vcw48m90ptyo) 16

[2.0 Summary Table of Input Information](#_ox1378is2i6k) 16

[3.0 Summary Table of Output Information](#_1tz8u8oyukac) 16

[3.1 Output Analysis of Rod Type A - ILP Model](#_bdzs2veevl1e) 16

[3.2 Output Analysis of Rod Type B - ILP Model](#_qxqngs4r2t8x) 17

[4.0 Integer Linear Programming Conclusions](#_azkt1h95slik) 18

[5.0 Comparing LP and ILP Outputs](#_nnfcre9ed130) 18

[5.1 Cost Comparisons of LP and ILP Models](#_iocp928jnr0s) 18

[5.2 Production Plan Comparisons of LP and ILP Models](#_23hqpj9wppz6) 19

[**Goal Programming (GP) Model**](#_27xn583f5343) **20**

[1.0 Introduction](#_qfadw6k75g3x) 20

[2.1 Goals for Type A metal rail](#_ea9pla3ys0lm) 20

[2.2 Goals for Type B metal rail](#_kefov2h2pohf) 20

[3.0 Summary Table of Input Information for GP](#_t3jke4fc6eiv) 21

[3.1 Input Information for Rod Type A](#_b9a763mx171b) 21

[3.2 Input Information for Rod Type B](#_fe62kih7n1k9) 22

[4.0 Summary Table of Output Information for GP](#_jxf1g9ua16lg) 23

[4.1 Output Analysis of Rod Type A](#_nnqco0z19fqq) 23

[4.2 Output Analysis of Rod Type B](#_lop9vx5mmy52) 24

[5.0 Summary of Sensitivity Analysis](#_2vn8b1itzmth) 25

[5.1 Sensitivity Analysis of Rod Type A](#_gv2n616twnfy) 25

[5.2 Sensitivity Analysis of Rod Type B](#_mpvlygbqlotp) 26

[6.0 Compare of LP and GP](#_hs8pkw7dcm7e) 27

[**Dynamic Programming (DP) Model**](#_1zzzccgrijq9) **29**

[1.0 Introduction](#_6xgxqbxg7uex) 29

[2.0 Input Information](#_r2g10pk8g083) 29

[3.0 Output Information for Dynamic Model](#_iorahhhqs31b) 29

[3.1 Output of Dynamic Model for Rod Type A](#_85o73pk9xnpr) 30

[3.2 Output of Dynamic Model for Rod Type B](#_9phmwag3j1jt) 31

[4.0 Conclusions of the Dynamic Model](#_87o8p9kugz7y) 31

[**Report to Management**](#_h80u8b6yaxk7) **32**

[1.0 Executive Summary](#_snn6x7zf3doy) 32

[2.0 Analysis](#_kon2gzlowihb) 32

[2.1 Linear Programming](#_7x23l368sc3l) 32

[2.2 Integer Linear Programming](#_8q1eqmim51qt) 33

[2.3 Goal Programming](#_rn66l4k5qu05) 33

[2.4 Dynamic Programming Model](#_qxdtk8v1xqs5) 34

[3.0 Conclusion & Recommendations](#_ylhcjjeiclmp) 34

[**Appendices**](#_95v2hc4rws53) **36**

[Appendix A: Original Problem, Assumptions, Background Information](#_fb1a65h2gbzl) 36

[Appendix A.1 Problem Statement](#_7g518eox3j5i) 36

[Appendix A.2 Modeling Assumptions](#_5o2oyzxolg9t) 37

[Appendix A.3 Problem Source](#_45m4qtfxpiti) 37

[Appendix A.4 Software Used](#_jzjexjn5dcwq) 37

[Appendix B: Detailed Explanation of Problem and Formulation for LP](#_o48vwgjx6pj4) 38

[Appendix B.2 Detailed Explanation of Formulation](#_yazzmc941kw3) 39

[Appendix C: AMPL for LP Model](#_7fzuw0s070bs) 41

[Appendix C.1 Rod Type A LP Model - AMPL Input](#_9976196uvbuz) 41

[Appendix C.2 Rod Type B LP Model - AMPL Input](#_ca9e0mx8uvua) 42

[Appendix C.3 AMPL Output for Rod Type A LP Model](#_yophcjqmo2v) 43

[Appendix C.4 AMPL Output for Rod Type B LP Model](#_awxm5mk1a1kj) 44

[Appendix D: LP Sensitivity Analysis](#_9012t2tsks03) 45

[Appendix D.1 Sensitivity Analysis for Rod A LP](#_cg7fwzttelze) 45

[Appendix D.2 Sensitivity Analysis for Rod B LP](#_jfsk2z5xk9vv) 46

[Appendix D.3 Constraint Sensitivity Analysis for Rod B Calculations](#_cfs73b5jsg13) 47

[Decreases to 49 inch part](#_ltsgczdp1uu) 47

[Increases to 49 inch part](#_flkah3j7mlsv) 47

[Note on sensitivity analysis](#_9e2mm8vojzo) 47

[Appendix E: AMPL for ILP Model](#_gbodkjyls9ud) 48

[Appendix E.1 Rod Type A ILP Model - AMPL Input](#_1was2mru5yhs) 48

[Appendix E.2 Rod Type B ILP Model - AMPL Input](#_deut9zx73c9t) 49

[Appendix E.3 AMPL Output for Rod Type A ILP Model](#_tcztk66s4bvo) 50

[Appendix E.3 AMPL Output for Rod Type B ILP Model](#_j5pnd3i3yvqd) 50

[Appendix F: Details of GP Model Formulation](#_oun5magq2mtc) 51

[Formulation for Rod type A GP Model:](#_x9nw3ejp80t3) 51

[Formulation for Rod type B GP Model:](#_8vfgdg7y9o5v) 52

[Appendix G1: AMPL for Type A GP Model](#_crcsempskwai) 54

[Appendix G2: AMPL output for Type A GP Model](#_659l8b4zkaw7) 56

[Appendix G3: SA for Type A GP Model](#_a9z3dprynh0i) 59

[Appendix G4: AMPL for Type B GP Model](#_hf73vqpqchb5) 67

[Appendix G5: AMPL output for Type B GP Model](#_qt1wdq45p24d) 69

[Appendix G6: SA for Type B GP Model](#_jro6vqlbwux7) 71

[Appendix H: Details of Dynamic Programming Model](#_vk86pauuzun8) 77

[Appendix I: AMPL for Dynamic Programming Model](#_hxnqnxbgea00) 78

[Appendix I.1 AMPL for Dynamic Programming Model Input](#_57el3vvkwjsn) 78

[Data File for Rod A](#_ksox9na8j45t) 78

[Data File for Rod B](#_qm75qztnj80o) 78

[Model File for Rods A & B](#_l7xlu1as707b) 79

[Run Script for Rods A & B](#_yeebg3cvxkoc) 81

[Appendix I.2 AMPL for Dynamic Programming Model Output](#_schyl0r1pqft) 83

[Output for Dynamic Programming Model Rod Type A](#_qvvsg9f7uhaa) 83

[Output for Dynamic Programming Model Rod Type](#_59l1jjt10nlj) 84

[Appendix I.3 Supplementary ILP Model from DP Model](#_vk89gyorkeme) 85

[AMPL Input](#_5xogehdf0lc4) 85

[AMPL Output](#_hpvnhgbi4t6j) 86

[Appendix J: List of Variables](#_c9f45xjza3e9) 87

[Appendix K: Problems & Suggestions for Future](#_h73ginnc79uc) 88

[Appendix K.1 Problems](#_7vzf6k4vgfg0) 88

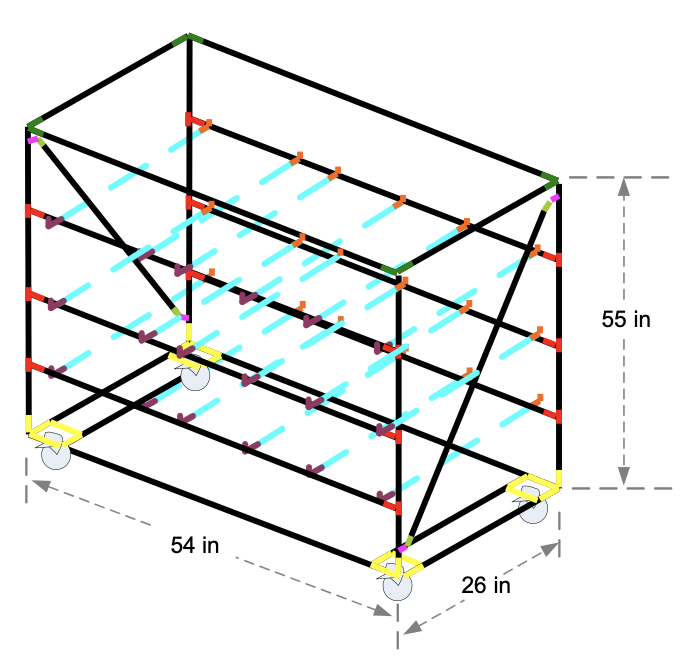
[Appendix K.2 Suggestions](#_wxf2w467hdzg) 88

[Appendix L: Project Evaluation](#_q3vh16p7l3dy) 89

# 

# Project Overview

The demand planning team has forecasted a large increase in demand and will start ramping up production to meet demand by next quarter. In preparation for increased production, Mr. Avila, the Los Angeles plant’s production manager, estimates that his team will need 5,000 metal baskets to support material handling in the facility. Mr. Avila has been in close collaboration with the special projects department to ensure that the material handling units are available in time.



***Figure 1: Metal basket engineering drawing***

The objective of this initiative is to minimize the overall production cost of producing 5,000 baskets. The final products will be used for material handling purposes in the factory that Mr. Avila manages. Mr. Avila has provided the detailed bill of materials (Section 1, Table 1.1) and the engineering drawings used in the baskets’ assembly(Figure1).

| Part | Base | | Sides | | Levels | Upper Frame | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Piece | Length | Width | Height | Supports | Length | Length | Width |
| Qty. | 2 | 4 | 4 | 2 | 6 | 2 | 2 |
| Inches | 51 | 24.5 | 55 | 50 | 49 | 49 | 22.5 |

***Table 1.1: Metal bar dimension and quantities***

The supply chain team has narrowed down two possible suppliers for raw materials. The first supplier can provide 184 inch metal rods at a cost of $9.50 per rod and the second supplier can provide 214 inch metal rods at a cost of $11.00 per rod. In addition to the cost of the raw materials, there is a cost associated with the scrap material of each rod of $0.02 per inch (Table 1.2). The same cost applies to any pieces produced but are unused in the final assembly of the 5,000 baskets.

|  | **Rod Type A** | **Rod Type B** | **Scrap** |
| --- | --- | --- | --- |
| Length (inches) | 184.00 | 214.00 | - |
| Cost ($) | 9.50 | 11.00 | 0.02 |

***Table 1.2: Types of metal rails and cost***

Although the overarching objective is to minimize the cost, some constraints were set by the factory due to certain physical and logical limitations:

* The machine used for rod cutting has only 5 knives. As a result, pieces can only be cut at most 5 times into 6 total pieces.
* Since each cutting pattern requires several minutes to set up, the production manager wants to reduce the number of changeovers in each shift. The production manager has agreed to a maximum of 8 total cutting patterns.
* The supply chain team can only get the $9.50 price for rod type A and the $11.00 price for rod type B if they agree to contract with that supplier exclusively. Therefore, we must only select either all of rod type A or all of rod type B.

In simple terms the goal of the project is to:

* Determine which rod to procure (either A or B)
* Provide appropriate production plan of cutting patterns for the chosen rod
* Meet the demand of parts according to the bill of materials to produce 5,000 baskets
* Minimize the total production cost which in effect is to minimize the total waste of scrap and unused materials

To determine the best course of action for producing the required metal baskets, the special projects team has developed multiple mathematical models with the hope of optimizing the process. The operations research models analyzed are the traditional linear programming (LP) model, an integer linear programming (ILP) model, a goal programming (GP) model, and finally a dynamic programming model. Each model is discussed in depth and the results are compared in our analysis. Please see the Executive Report section for the final conclusions and recommendations.

# 

# Linear Programming (LP) Model

## 1.0 Introduction

In the given problem, we understand that the relationships between the involved entities are linear. The objective of our problem, for instance, is to minimize the cost of producing 5000 baskets and it can be expressed as a linear combination of the cost of buying raw material and removal of waste/unused material. Similarly, the requirement on the minimum number of parts of each dimension, are a linear combination of the number of rods of each pattern being cut. Hence, we choose to solve it using the Linear Programming method. For detailed explanation of the linear program formulation, please refer to Appendix B.

We start by looking at all possible patterns that can be cut, based on the dimensions of type A and type B rods, and selecting 8 patterns for each type. Our goal here is to determine how many rods of each type are to be purchased by the procurement team in order to fulfill this production order, while incurring as minimal cost as possible. The patterns were arbitrarily chosen with the only requirement being that no additional final pieces of any of the lengths in demand can be cut from the wasted material.

Given the limitation that only one type of rod (A or B) must be selected for the final production plan, we chose to separate the model into two models -- one with the objective of minimizing the production cost of rod A and the other with the object of minimizing the production cost of rod B. The motivation behind this is that, since both the LP model for rod A and rod B are considered “optimal” for the cutting patterns we determined, we would be able to compare the cost of the two to determine which rod type should be chosen.

While we are able to model this problem as a linear model, we expect that the solution will provide fractional values for each cutting pattern. For the final production model, it would not make practical sense to provide the procurement team with a plan to procure any fraction of a rod. As such, this model will be used as the foundation for other models such as integer linear programming, goal programming, and dynamic programming and will serve as a baseline for comparison.

## 2.0 Summary Table of Input Information

The below information in this section shows the data used as an input to the model. The primary input is the patterns that are used in cutting. These patterns for each rod type were chosen arbitrarily by using a condition that no additional final pieces can be obtained from the left over material or waste.

### 2.1 Input Information for Rod Type A

The summary tables below only summarize the relevant information about type A metal rail. The final row, “Min. Required” is the total demand for each part needed to construct all 5000 metal baskets. For example, for the 51 inch part, 2 pieces are needed for one basket so the total demand is 2 \* 500 = 10000 pieces. For all other information, please refer to Appendix for details.

| Type A metal rod with length of 184 inches | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | # of 51 in. parts | # of 24.5 in. parts | # of 55 in. parts | # of 50 in. parts | # of 49 in. parts | # of 22.5 in. parts | Extra in. wasted | Decision  Variable |
| 1 | 0 | 0 | 0 | 0 | 2 | 3 | 18.5 | X1A |
| 2 | 0 | 3 | 2 | 0 | 0 | 0 | 0.5 | X2A |
| 3 | 0 | 0 | 2 | 1 | 0 | 1 | 1.5 | X3A |
| 4 | 0 | 1 | 2 | 0 | 0 | 2 | 4.5 | X4A |
| 5 | 0 | 2 | 2 | 0 | 0 | 1 | 2.5 | X5A |
| 6 | 1 | 1 | 0 | 1 | 1 | 0 | 9.5 | X6A |
| 7 | 1 | 0 | 2 | 0 | 0 | 1 | 0.5 | X7A |
| 8 | 2 | 1 | 1 | 0 | 0 | 0 | 2.5 | X8A |
| Min.  Required | 2\*5000=10000 | 4\*5000=20000 | 4\*5000=20000 | 2\*5000=10000 | 8\*5000=40000 | 2\*5000=10000 |  |  |
| Objective: Minimize the total cost | | | | | | | | |

***Table 2.1.LP Cutting patterns and minimum demand for type A metal rail***

### 2.2 Input Information for Rod Type B

The summary tables below only summarize the relevant information about type B metal rail. For all other information, please refer to Appendix for details.

| Type B metal rod with length of 214 inches | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | # of 51 in. parts | # of 24.5 in. parts | # of 55 in. parts | # of 50 in. parts | # of 49 in. parts | # of 22.5 in. parts | Extra in. wasted | Decision  Variable |
| 1 | 0 | 0 | 0 | 0 | 4 | 0 | 18 | X1B |
| 2 | 0 | 2 | 0 | 1 | 1 | 0 | 15 | X2B |
| 3 | 0 | 0 | 3 | 0 | 1 | 0 | 0 | X3B |
| 4 | 0 | 1 | 3 | 0 | 0 | 1 | 2 | X4B |
| 5 | 0 | 0 | 0 | 3 | 1 | 0 | 15 | X5B |
| 6 | 1 | 0 | 2 | 1 | 0 | 0 | 3 | X6B |
| 7 | 1 | 2 | 2 | 0 | 0 | 0 | 4 | X7B |
| 8 | 1 | 2 | 1 | 0 | 0 | 2 | 14 | X8B |
| Min. Required | 2\*5000=10000 | 4\*5000=20000 | 4\*5000=20000 | 2\*5000= 10000 | 8\*5000=40000 | 2\*5000=10000 |  |  |
| Objective: Minimize the total cost | | | | | | | | |

***Table 2.2.LP Cutting patterns and minimum demand for type B metal rail***

# 

## 3.0 Summary Table of Output Information

### 3.1 Output Analysis of Rod Type A

The table below shows the output or decisions from the LP model for rod type A. It is important to note that the model had 8 possible cutting patterns but only the patterns 1,2,3,6,7 are included in the table because the remaining patterns 4 and 8 were not used by the model. The complete patterns and quantities of each part produced by the pattern can be found in ***Table 2.1.LP.*** Using the results from this model, the overall demand is met with a cost of $346,771.43.

The *Waste Cost per Rod ($)* column is calculated by multiplying the amount of waste produced by the pattern by the cost in $/inch of waste: $0.02/inch. For example for pattern 1, the waste cost per rod is 18.5 inches \* $0.02/inch = $0.37. The waste for each pattern can also be found in ***Table 2.1.LP.*** The *Waste Cost per Pattern ($)* column in the table is simply the quantity of each pattern produced times the *Waste Cost per Rod ($).* For pattern 1, this would be 16428.6 \* 0.37 = 6078.58.

| Cutting results | | | | | |
| --- | --- | --- | --- | --- | --- |
| Pattern i | Qty. | Waste Cost per Rod ($) | | Waste Cost per Pattern ($) | |
| 1 | 16428.6 | 0.37 | | 6078.58 | |
| 2 | 4285.71 | 0.01 | | 42.86 | |
| 3 | 2857.14 | 0.03 | | 85.71 | |
| 6 | 7142.86 | 0.19 | | 1357.14 | |
| 7 | 2857.14 | 0.01 | | 28.57 | |

***Table 3.1.LP Cutting results for Type A Rod***

### 

### 3.2 Output Analysis of Rod Type B

The table below shows the output or decisions from the LP model for rod type B. Unlike the ILP model for rod type A, each of the 8 cutting patterns were used by the model. The complete patterns and quantities of each part produced by the pattern can be found in the LP section in Table 2.1.LP. Using the results from this model, the overall demand is met with a cost of $267,200.00.

| Cutting results | | | | | |
| --- | --- | --- | --- | --- | --- |
| Pattern i | Qty. | Waste Cost per Rod ($) | | Waste Cost per Pattern ($) | |
| 1 | 9226.19 | 0.36 | | 3321.43 | |
| 4 | 1428.57 | 0.04 | | 57.14 | |
| 5 | 3095.24 | 0.30 | | 928.57 | |
| 6 | 714.286 | 0.06 | | 42.86 | |
| 7 | 5000 | 0.08 | | 400 | |
| 8 | 4285.71 | 0.28 | | 1200 | |

***Table 3.2.LP Cutting results for Type B rod***

## 4.0 Linear Programming Model Summary

The above tables list the optimal solution for each type for producing at least 5000 baskets. Type A requires a minimum cost of $346,771.43 and type B requires a minimum cost of $267,200. Given the models at this point, we choose type B with the following production plan and then do sensitivity analysis with type B: (further explanation is in the Appendix)

1. We plan to buy 9226.19 pattern 1 and use it to produce 36904.76 bars with 50 inches. The purchase cost is $87648.81.
2. We plan to buy 1428.57 pattern 4 and use it to produce 1428.57 bars with 24.5 inches, 4285.71 bars with 55 inches, 1428.57 bars with 22.5 inches. The purchase cost is $13571.42.
3. We plan to buy 3095.24 pattern 5 and use it to produce 9287.72 bars with 50 inches and 3095.24 bars with 49 inches. The purchase cost is $29404.78.
4. We plan to buy 714.286 pattern 6 and use it to produce 714.286 bars with 51 inches, 1428.572 bars with 55 inches and 714.286 bars with 50 inches. The purchase cost is $6785.717.
5. We plan to buy 5000 pattern 7 and use it to produce 5000 bars with 50 inches, 10000 bars with 24.5 inches and 10000 bars with 55 inches. The purchase cost is $47500.
6. We plan to buy 4285.71 pattern 8 and use it to produce 4285.71 bars with 51 inches, 8571.42 bars with 24.5 inches, 4285.71 bars with 55 inches and 8571.42 bars with 22.5 inches. The purchase cost is $40714.25.
7. The total cost will be $267200, consisting of $261249.956 purchase cost and $5692.857 from excess bars.

## 5.0 Linear Programming Sensitivity Analysis

Based on the results of the linear programming model from the previous section, Rod B proves to be more desirable in terms of the lowest production cost. As such, we will perform sensitivity analysis for only the Rod B LP model. This sort of analysis seeks to understand how cost might change if certain things in the model were to change. It enables us to understand the model at a deeper level and ask what if style questions without having to solve the model multiple times. To do this analysis, we will perform sensitivity analysis on both the variables and the constraints of the model. It is important to note that because our model is dependent on the cutting patterns, there is an interrelationship between variables wherein a variable or constraint that increases or decreases the entire model could be affected because the cutting patterns include multiple rod lengths. If each pattern was independent in the sense that pattern X only produced piece Y, there would be no correlation between variables or constraints which would make the sensitivity analysis more meaningful. However, since in our case pattern X can produce pieces Y, Z, etc., this is not the case. Nonetheless, sensitivity analysis will be performed to see what useful information we can gain from the model without solving it again.

### 5.1 Rod B Sensitivity Analysis of Variables

In this linear programming model, we have a total of 8 variables that represent the 8 cutting patterns. While we cannot analyze all 8 variables using sensitivity analysis, we will analyze the variable with the largest total contribution to the objective function. The goal of this analysis is to see how a change to the objective function coefficient would change the model.

Given the results for Rod B LP in table 3.3.LP above, it is clear that pattern 1 (variable X1B) has the largest contribution to the objective function (total cost equation) because it uses the highest amount of rods where each rod used costs the same amount. As mentioned above, there is an implicit interrelationship between the variables because each variable has the same objective function coefficient which means that for any change in the cost per rod, we must ensure that the changes are feasible for all variables.

The current objective function coefficient for pattern 1 is $15.28 which factors in the cost of the rod and other associated production costs. For pattern 1 with the largest contribution to the objective function, the allowable range for the cost per rod is between $0.00 and $15.28. However, by taking the maximum value from the lower end of the range for all patterns, it would only be logical to go down to $15.28. Conversely, by taking the minimum from the higher range for all patterns, it would only be logical to go up to $15.28. Thus there is no room for changes to any objective function coefficients.

Since there can be no practical change in objective function coefficients, we will not bother by creating a sensitivity graph for this analysis since the graph would just consist of a single point. Should the cost change from the $11 per rod or if the cost of the waste were to change from $0.02 per inch, we would need to re-solve the problem to determine the effects on the model. Clearly, however, the total production cost is directly proportional to the cost of the rods and cost of waste where if the costs increase, the production cost increase or if the costs decrease, the production costs decrease. We do not know what the effect on the model would be if rod cost increases and waste cost decreases or vice versa.

In addition to the highest contributing variable to the objective function, we will also look at the variable with the smallest reduced cost. Since variable X2B is the only variable with reduced cost not equal to zero, we know that this is the only non-basic variable meaning that the number of rods using pattern 2 is zero. For the model to decide to use pattern 2, the objective function coefficient must increase by $2.80. For the same reasons mentioned above, it is not feasible to increase the objective function coefficients due to the interrelationship of variables. As a result, we are not able to determine the effect on the model.

### 

### 5.2 Rod B Sensitivity Analysis of Constraints

The objective of the sensitivity analysis on the constraints is to determine how much the objective function would increase or decrease given any changes to the right hand side (RHS) of a constraint. In simple terms, if demand for a specific part increased or decreased by X amount of pieces, how much would the total cost change? This could be feasible for example if the design engineering team were to come up with a new design that might use two 55 inch parts instead of the current requirement of 4 or perhaps if the production team determined that they need more or less than 5000 metal baskets. This is because the calculation for the right hand side of the constraint is determined by the quantity of a part required for 1 basket multiplied by the total number of baskets needed, 5000.

For our purposes all constraints can be considered binding constraints because slack or excess is exactly or very close to zero. This means that we are meeting all of the constraints for demand without any shortages in material or without ordering too much material. The shadow price of a constraint tells us how much the objective function would increase or decrease if the right hand side of the constraint were to increase by 1 unit. Please see the shadow price for each constraint in the table below.

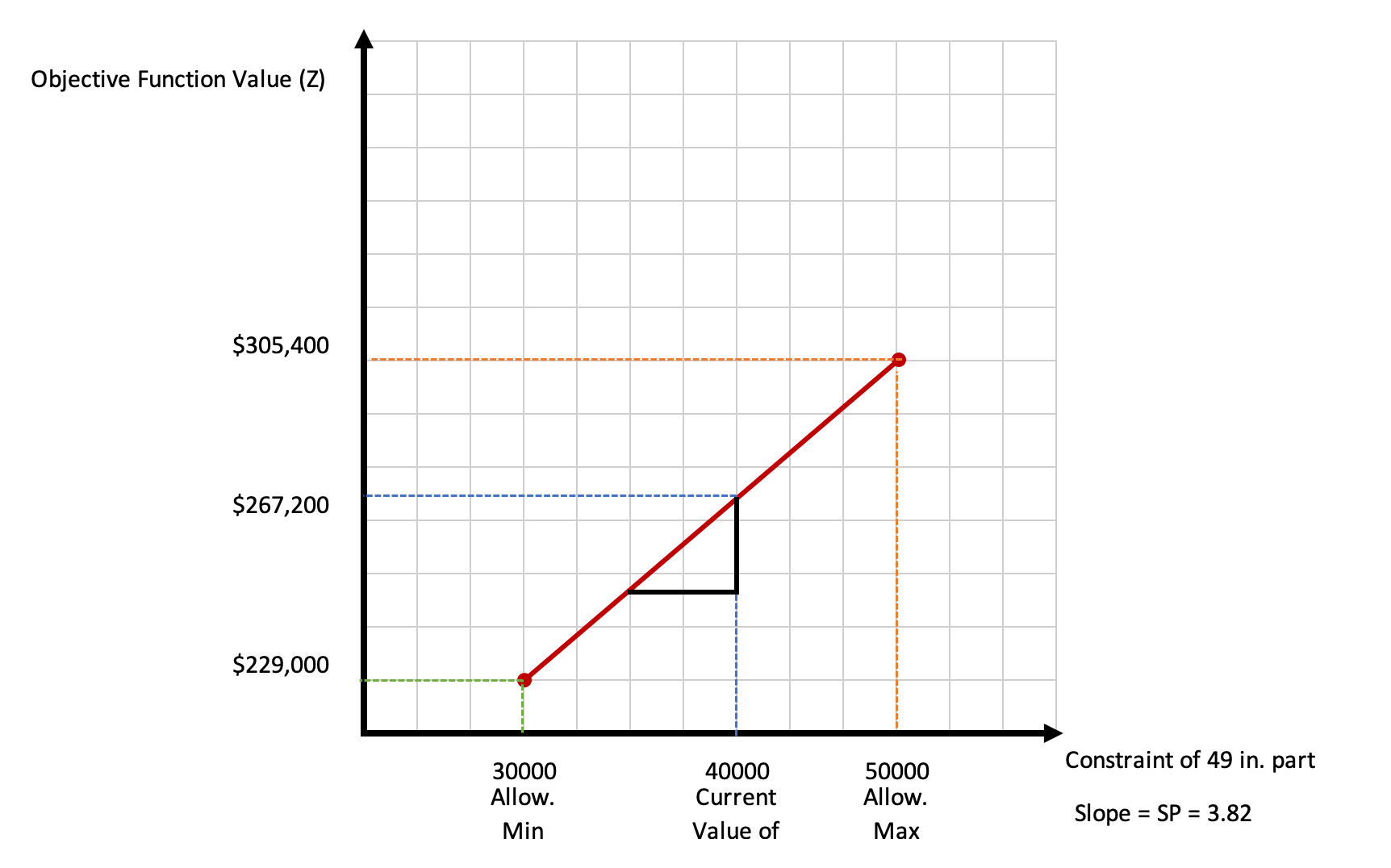
| **Piece (inches) - Constraint** | **Demand for 1 Unit** | **Min. Allowable RHS** | **Current RHS - Total Demand** | **Max. Allowable RHS** | **Shadow Price** |
| --- | --- | --- | --- | --- | --- |
| 51 | 2 | 9000 | 10000 | 12500 | +3.82 |
| 24.5 | 4 | 10000 | 20000 | 21428.6 | +1.91 |
| 55 | 4 | 15000 | 20000 | 50000 | +3.82 |
| 50 | 2 | 714.286 | 10000 | 120714 | +3.82 |
| 49 | 8 | 3095.24 | 40000 | Infinite | +3.82 |
| 22.5 | 2 | 0 | 10000 | 20000 | +1.91 |

***Table 5.1.LP Constraint Sensitivity Analysis for Rod B***

The minimum and maximum allowable ranges in the table above are the amount that the RHS for the given constraint can change by. As a result, we can calculate what would happen to the objective function if the RHS changed to any number within the range. Any number outside of the range would require solving the problem again.

To analyze how the model will be affected, we will assume a hypothetical scenario where the design engineering team has just developed a prototype for a new basket that would change the requirements of the bill of materials. The design engineers have tested this new design and have determined that they can create the basket to transport the same amount of material by using two less 49 inch parts thus making the new demand per unit 6. Scaling this change to the 5000 units, this would mean that the total required number of 49 inch parts would go from 40,000 pieces to 30,000 pieces. By making this change, the resulting improvement to the total cost (objective function) would go from $267,200 to $229,000.

Although the allowable range for 49 inch parts goes from around 3000 to infinity, we will assume that increases or decreases can only happen in increments of 2 parts per unit. This logic is assumed to apply to all parts where the maximum increase/decrease is ±2. For the 49 inch piece the new range for all baskets is 30,000 - 50,000 pieces where the current value of the RHS is at 40,000 pieces. The chart below shows the general relationship between increasing and decreasing the number of 49 inch pieces and its effect on the objective function. The calculations can be found in Appendix D.3.



***Figure 5.1.LP - Sensitivity Chart for 49 inch piece constraint***

# 

# Integer Linear Programming (ILP) Model

## 1.0 Introduction

The previous linear programming (LP) models for rods A and B involved optimizing the number of rods of each type needed to fulfill the demand to build 5000 metal baskets. This objective of the LP model was to minimize the cost of procurement, cost of waste, and cost of unused parts. While a solution to the LP modes for both rod types was attained, the results of how many rods of each type to use on a specific cutting pattern were not integer values.

Based on the contracts that the procurement team created with the suppliers for rods A and B, the company is not able to procure fractional components of a rod so the standard lengths of 184 inches or 214 inches must be purchased. To provide a more accurate presentation on how many rods to purchase of each type, the team expanded upon the linear programming models to develop an integer linear programming (ILP) model. While the objective remains the same for the ILP model, the variables must now strictly be integers greater than or equal to zero. Using this method, the exact number of rods needed can be provided to the supply chain team to make the purchase requisition.

## 2.0 Summary Table of Input Information

For the integer linear programming model, the only change from the standard linear programming model is that the variables are restricted to integer values only. As such, the input information given in section 2.0 of the linear programming segment of this report is identical to the input information for the ILP model. Please refer to 2.0 in the linear programming section, tables 2.1.LP and 2.2.LP.

The full model formulation written in AMPL can be found in Appendix E.

## 3.0 Summary Table of Output Information

Since the ILP model output is only integer values, the overall cost is as it is expected to be - higher than the linear programming model. This makes sense because for example, where 1333.33 rods for a given cutting pattern would have fulfilled the demand, you will now have 1334 rods which will add cost from the overall price of the rod and the waste of that rod.

Below are the results from the ILP models for both rod type A and rod type B.

### 3.1 Output Analysis of Rod Type A - ILP Model

The table below shows the output or decisions from the integer LP model for rod type A. It is important to note that the model had 8 possible cutting patterns but only the patterns 1,2,3,6,7 are included in the table because the remaining patterns 4 and 8 were not used by the model. The complete patterns and quantities of each part produced by the pattern can be found in the LP section in ***Table 2.1.LP.*** Using the results from this model, the overall demand is met with a cost of $346,778.96.

The calculations for *Waste Cost per Rod ($)* and *Waste Cost per Pattern ($)* are the same as previously mentioned in LP section 3.1.

| Cutting results | | | | | |
| --- | --- | --- | --- | --- | --- |
| Pattern i | Qty. | Waste Cost per Rod ($) | | Waste Cost per Pattern ($) | |
| 1 | 16429 | 0.37 | | 6078.73 | |
| 2 | 4286 | 0.01 | | 42.83 | |
| 3 | 2857 | 0.03 | | 85.71 | |
| 6 | 7143 | 0.19 | | 1357.17 | |
| 7 | 2857 | 0.01 | | 28.57 | |

***Table 3.1.ILP Cutting results for Type A Rod***

### 3.2 Output Analysis of Rod Type B - ILP Model

The table below shows the output or decisions from the integer LP model for rod type B. Unlike the ILP model for rod type A, each of the 8 cutting patterns were used by the model. The complete patterns and quantities of each part produced by the pattern can be found in the LP section in Table 2.1.LP. Using the results from this model, the overall demand is met with a cost of $267,214.26

The calculations for *Waste Cost per Rod ($)* and *Waste Cost per Pattern ($)* are the same as previously mentioned in LP section 3.1.

| Cutting results | | | | | |
| --- | --- | --- | --- | --- | --- |
| Pattern i | Qty. | Waste Cost per Rod ($) | | Waste Cost per Pattern ($) | |
| 1 | 9226 | 0.36 | | 3321.36 | |
| 2 | 1 | 0.30 | | 0.30 | |
| 4 | 1428 | 0.04 | | 57.12 | |
| 5 | 3095 | 0.30 | | 928.5 | |
| 6 | 715 | 0.06 | | 42.9 | |
| 7 | 5000 | 0.08 | | 400 | |
| 8 | 4286 | 0.28 | | 1200.08 | |

***Table 3.2.ILP Cutting results for Type B rod***

## 4.0 Integer Linear Programming Conclusions

While the total cost of the ILP model is more than the LP model, the difference between the two is nominal. This implies that the integer linear programming model makes more practical sense in terms of providing an overall production plan for building the 5000 metal baskets required. This is due to the fact that the integer programming model provides the exact requirements for the number of rods cut using each pattern which effectively removes guesswork or rounding from the process. The comparisons between both LP and ILP models can be found in the section below.

The AMPL inputs and outputs of the ILP models for rod types A and B can be found in Appendix E.

## 5.0 Comparing LP and ILP Outputs

### 5.1 Cost Comparisons of LP and ILP Models

|  | Total Cost | | |
| --- | --- | --- | --- |
| Rod | LP | ILP | Difference |
| Type A | $346,771.43 | $346,778.96 | $7.53 |
| Type B | $267,200.00 | $267,214.26 | $14.26 |

***Table 5.1.ILP Cost Comparisons***

The table above shows the total cost objectives for LP and ILP models. From this we can see that Rod Type B is clearly the best option to produce the 5000 metal baskets as it results in a lower total cost than Rod Type A. Furthermore, between the LP and ILP models, the ILP model should be selected since the difference between the cost is nominal. Therefore using these two models, the ILP model for Rod Type B should be used to generate the production plan for building the metal baskets. In subsequent sections, two additional models will be explored to see if better results can be achieved.

### 5.2 Production Plan Comparisons of LP and ILP Models

| **Cutting Result Quantities** | | | | | | |
| --- | --- | --- | --- | --- | --- | --- |
| **Pattern i** | **Rod A - LP** | **Rod A - ILP** | | **Rod B - LP** | | **Rod B - ILP** |
| 1 | 16,428.6 | 16,429 | | 9,226.19 | | 9,226 |
| 2 | 4,285.71 | 4,286 | | 0 | | 1 |
| 3 | 2,857.14 | 2,857 | | 0 | | 0 |
| 4 | 0 | 0 | | 1,428.57 | | 1,428 |
| 5 | 0 | 0 | | 3,095.24 | | 3,095 |
| 6 | 7,142.86 | 7,143 | | 714.286 | | 715 |
| 7 | 2,857.14 | 2,857 | | 5,000 | | 5000 |
| 8 | 0 | 0 | | 4285.71 | | 4,286 |
| Total qty. | 33,571.45 | 33,572 | | 23,749.996 | | 23,751 |
| Total in. | 6,177,146.8 | 6,177,248 | | 5,082,499.144 | | 5,082,714 |
| Total Cost | $346,771.43 | $346,778.96 | | $267,200.00 | | $267,214.26 |

***Table 5.2.ILP Cost Cutting patterns***

Based on the production plan, we can see that the lower cost of rod type B for both LP and ILP models is confirmed since the overall amount of rods necessary to fulfill the demand is fewer. Since rods A and B are different lengths, adjusting the number of rods to inches overall, where rod A is 184 inches long and rod B is 214 inches long, the ILP model for rod B actually requires around 1 million inches of material less than the ILP model for rod A. The cutting patterns for Rod A can be found in LP section 2.1 and the cutting patterns for Rod B can be found in LP section 2.2. The same cutting patterns apply for both LP and ILP models.

# Goal Programming (GP) Model

## 1.0 Introduction

In the Linear Programming and Integer Linear Programming Sections, we considered rigid constraints. However, we want to relax some of these constraints to instead be goals. In order to see how the production plan might change. We want to investigate goals related to meeting demand of 51 inch pieces constraint, meeting demand of 24.5 inch pieces constraint, and meeting some cost level. To analyze this further, our team develop the following goals:

**2.0 Goals for GP**

### 2.1 Goals for Type A metal rail

1. The first priority Goal 1 (G1): Total waste cost from extra pieces wasted at most $7592.9
2. The second priority Goal 2 (G2): The number of 51 inch pieces should be at least 10000
3. The third priority Goal 3 (G3): The number of 24.5 inch pieces should be at least 20000

### 2.2 Goals for Type B metal rail

1. The first priority Goal 1 (G1): Total waste cost from extra pieces wasted at most $5692.9
2. The second priority Goal 2 (G2): The number of 51 inch pieces should be at least 10000
3. The third priority Goal 3 (G3): The number of 24.5 inch pieces should be at least 20000

The reasons for those goals are in Appendix A.2.

### 

## 3.0 Summary Table of Input Information for GP

### 3.1 Input Information for Rod Type A

The summary tables below only summarize the relevant information about type A metal rail. For all other information, please refer to Appendix for detail.

| Type A metal rail with length of 184 inches | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | # of 51 in. parts | # of 24.5 in. parts | # of 55 in. parts | # of 50 in. parts | # of 49 in. parts | # of 22.5 in. parts | Extra in. wasted | Decision  Variable |
| 1 | 0 | 0 | 0 | 0 | 2 | 3 | 18.5 | X1A |
| 2 | 0 | 3 | 2 | 0 | 0 | 0 | 0.5 | X2A |
| 3 | 0 | 0 | 2 | 1 | 0 | 1 | 1.5 | X3A |
| 4 | 0 | 1 | 2 | 0 | 0 | 2 | 4.5 | X4A |
| 5 | 0 | 2 | 2 | 0 | 0 | 1 | 2.5 | X5A |
| 6 | 1 | 1 | 0 | 1 | 1 | 0 | 9.5 | X6A |
| 7 | 1 | 0 | 2 | 0 | 0 | 1 | 0.5 | X7A |
| 8 | 2 | 1 | 1 | 0 | 0 | 0 | 2.5 | X8A |
| Min.  Required | 2\*5000=10000 | 4\*5000=20000 | 4\*5000=20000 | 2\*5000=10000 | 8\*5000=40000 | 2\*5000=10000 |  |  |
| Goal 1: Total waste cost from extra pieces wasted at most $7592.9 | | | | | | | | |
| Goal 2: The number of 51 inch pieces should be at least 10000 | | | | | | | | |
| Goal 3: The number of 24.5 inch pieces should be at least 20000 | | | | | | | | |

***Table 3.1.GP Cutting patterns and minimum demand for type A metal rail***

### 3.2 Input Information for Rod Type B

The summary tables below only summarize the relevant information about type B metal rail. For all other information, please refer to Appendix for details.

| Type B metal rod with length of 214 inches | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | # of 51 in. parts | # of 24.5 in. parts | # of 55 in. parts | # of 50 in. parts | # of 49 in. parts | # of 22.5 in. parts | Extra in. wasted | Decision  Variable |
| 1 | 0 | 0 | 0 | 0 | 4 | 0 | 18 | X1B |
| 2 | 0 | 2 | 0 | 1 | 1 | 0 | 15 | X2B |
| 3 | 0 | 0 | 3 | 0 | 1 | 0 | 0 | X3B |
| 4 | 0 | 1 | 3 | 0 | 0 | 1 | 2 | X4B |
| 5 | 0 | 0 | 0 | 3 | 1 | 0 | 15 | X5B |
| 6 | 1 | 0 | 2 | 1 | 0 | 0 | 3 | X6B |
| 7 | 1 | 2 | 2 | 0 | 0 | 0 | 4 | X7B |
| 8 | 1 | 2 | 1 | 0 | 0 | 2 | 14 | X8B |
| Min. Required | 2\*5000=10000 | 4\*5000=20000 | 4\*5000=20000 | 2\*5000= 10000 | 8\*5000=40000 | 2\*5000=10000 |  |  |
| Goal 1: Total waste cost from extra pieces wasted at most $5692.9 | | | | | | | | |
| Goal 2: The number of 51 inch pieces should be at least 10000 | | | | | | | | |
| Goal 3: The number of 24.5 inch pieces should be at least 20000 | | | | | | | | |

***Table 3.2.GP Cutting patterns and minimum demand for type B metal rail***

### 

### 

## 4.0 Summary Table of Output Information for GP

### 4.1 Output Analysis of Rod Type A

| Decision Variable and Goal | Solution value |
| --- | --- |
| X1A | 15572 |
| X2A | 8570 |
| X3A | 1144 |
| X6A | 8856 |
| X8A | 572 |
| S3P | 15138 |
| Min. Z1 | 0 |
| Min. Z2 | 0 |
| Min. Z3 | 0 |

***Table 4.1.GP results for Type A Rod***

The table above shows the output from the GP model for rod type A. The original output is located at ***Appendix G2: AMPL output for Type A GP Model***. It is important to note that the model had 8 possible cutting patterns but only the patterns 1,2,3,6,8 are included in the table because the remaining patterns were not used by the model. Using the results from this model, all those three goals can be achieved.

According to our model, we are going to use 15572 type A bars for pattern 1, 8570 type A bars for pattern 2, 1144 type A bars for pattern 3, 8856 type A bars for pattern 6, and 572 type A bars for pattern 8. Z1 = 0. This is related to waste cost from extra pieces wasted and indicates there is no undesirable situation. The waste cost from extra pieces wasted is exactly $7592.9. Z2 = 0. This is related to the number of 51 inch pieces and indicates there is no undesirable situation. The number of 51 inch pieces is exactly 10000. Z3=0. This is related to the number of 24.5 inch pieces and indicates there is no undesirable situation. The number of 24.5 inch pieces is 35138, which is higher than our goal by 15138, and this amount is the value of S3P, which is desirable.

### 

### 4.2 Output Analysis of Rod Type B

| Decision Variable and Goal | Solution value |
| --- | --- |
| X2B | 5000 |
| X3B | 24357 |
| X4B | 10000 |
| X5B | 10643 |
| X6B | 10000 |
| Min. Z1 | 0 |
| Min. Z2 | 0 |
| Min. Z3 | 0 |

***Table 4.2.GP results for Type B Rod***

The table above shows the output from the GP model for rod type B. The original output is located at ***Appendix G4: AMPL output for Type B GP Model***. It is important to note that the model had 8 possible cutting patterns but only the patterns 2,3,4,5,6 are included in the table because the remaining patterns were not used by the model. Using the results from this model, all the three goals can be achieved.

According to our model, we are going to use 5000 type B bars for pattern 2, 24357 type B bars for pattern 3, 10000 type B bars for pattern 4, 10643 type B bars for pattern 4, 10000 type B bars for pattern 5.

We obtained Z1 = 0. This is related to waste cost from extra pieces wasted and indicates there is no undesirable situation. The cost from extra pieces wasted is exactly $5692.9.

We also obtained Z2 = 0 and Z3 = 0. These are related to the number of 51 inch pieces and 24.5 inch pieces respectively. The value 0 indicates that there is no undesirable situation. The number of 51 inch pieces is exactly 10000 and the number of 24.5 inch pieces is exactly 20000.

## 5.0 Summary of Sensitivity Analysis

### 5.1 Sensitivity Analysis of Rod Type A

|  | **Priorities** | | |
| --- | --- | --- | --- |
| **Trial** | **Highest** | **Second Highest** | **Lowest** |
| 1 | S1P | S2N | S3N |
| 2 | S2N | S1P | S3N |
| 3 | S2N | S3N | S1P |

|  | **Optimal Solution** | | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Variables** | | | | | | **Undesirable Deviation** | | |
| **Trial** | **X1A** | **X2A** | **X3A** | **X6A** | **X7A** | **X8A** | **Z1** | **Z2** | **Z3** |
| 1 | 15572 | 8570 | 1144 | 8856 | 0 | 572 | 0 | 0 | 0 |
| 2 | 15572 | 8570 | 1144 | 8856 | 0 | 572 | 0 | 0 | 0 |
| 3 | 16427.9 | 4285.26 | 2858.95 | 7144.21 | 2855.79 | 0 | 0 | 0 | 0 |

***Table 5.1.GP Sensitivity Analysis for Type A Rod***

Analysis of the above table indicates that the solution will remain the same if the priority of the highest and the next highest goal are switched (Trial 2).

However, if the number of 51 inch pieces has the highest priority, followed by the number of 24.5 inch pieces, and finally the waste cost from extra pieces wasted (Trial 3), then the solution will change. In this case, it is suggested to use 16427.9 type A bars for pattern 1, 4285.26 type A bars for pattern 2, 2858.95 type A bars for pattern 3, 7144.21 type A bars for pattern 6, and 2855.79 type A bars for pattern 7 . This case will result in $7592.9 waste cost from extra pieces wasted. The number of 51 inch pieces is exactly 10000. The number of 24.5 inch pieces is 20000. All these three goals are exactly matched, and this can be observed by the value of all deviation variables being 0.

Since all of our three goals can be achieved, trial 1 and 2 can provide 15138 extra 24.5 inch pieces by having the same waste cost from extra pieces with trial 3, which is desirable. Therefore, the final suggestion is to consider the original priority and solution.

### 5.2 Sensitivity Analysis of Rod Type B

|  | **Priorities** | | |
| --- | --- | --- | --- |
| **Trial** | **Highest** | **Second Highest** | **Lowest** |
| 1 | S1P | S2N | S3N |
| 2 | S2N | S1P | S3N |
| 3 | S2N | S3N | S1P |

|  | **Optimal Solution** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Variables** | | | | | **Undesirable Deviation** | | |
| **Trial** | **X2B** | **X3B** | **X4B** | **X5B** | **X6B** | **Z1** | **Z2** | **Z3** |
| 1 | 5000 | 24357 | 10000 | 10643 | 10000 | 0 | 0 | 0 |
| 2 | 5000 | 24357 | 10000 | 10643 | 10000 | 0 | 0 | 0 |
| 3 | 5000 | 24357 | 10000 | 10643 | 10000 | 0 | 0 | 0 |

***Table 5.2.GP Sensitivity Analysis for Type B Rod***

Analysis of the above table indicates that the solution will remain the same regardless of the order of the goals. This means that regardless of how we prioritise one goal over the other, we will achieve the same results.

As we can notice in section 4.2 and trial one here, the obtained solution meets all of our goals exactly with no deviations and fulfills all the constraints as well. Therefore, changing the order of the goals has no effect on the solution.

### 

### 

## 6.0 Compare of LP and GP

***6.1 Compare of LP and GP for Rod Type A***

|  | Solution value for LP | Solution value for GP |
| --- | --- | --- |
| X1A | 16428.6 | 15572 |
| X2A | 4285.71 | 8570 |
| X3A | 2857.14 | 1144 |
| X6A | 7142.86 | 8856 |
| X7A | 2857.14 | 0 |
| X8A | 0 | 572 |
| Total Cost | $346,771.43 | $354,412.9\* |
| Waste Cost for Extra Pieces | $7592.9 | $7592.9 |

\* For the total cost of the GP section, we use the similar formula like what we did in the LP section, but we delete the cost of unused pieces of 51 and 24.5 inches, since based on our goal in GP, these two kinds of pieces are desirable. We want as much as we can produce. So we think they should not be considered as unused waste.

***Table 6.1. LP and GP results for Type A Rod***

The above tables list the optimal solution for LP and GP for Rod Type A. For the LP section, the total minimum cost is $346,771.43. For the GP section, the total minimum cost is $354412.9. Both LP and GP sections have waste cost for extra pieces of $7592.9

For the LP section, we are going to use 16428.6 type A bars for pattern 1, 4285.71 type A bars for pattern 2, 2857.14 type A bars for pattern 3, 7142.86 type A bars for pattern 6, and 2857.14 type A bars for pattern 7. The total number of type A metal rails we need to buy is 33571.45. For the GP section, we are going to use 15572 type A bars for pattern 1, 8570 type A bars for pattern 2, 1144 type A bars for pattern 3, 8856 type A bars for pattern 6, and 572 type A bars for pattern 8. The total number of type A metal rails we need to buy is 34714.

Both LP and GP will let us produce 5000 baskets, and the LP method will provide a smaller total minimum cost for type A rod.

***6.2 Compare of LP and GP for Rod Type B***

|  | Solution value for LP | Solution value for GP |
| --- | --- | --- |
| X1B | 9226.19 | 0 |
| X2B | 0 | 5000 |
| X3B | 0 | 24357 |
| X4B | 1428.57 | 10000 |
| X5B | 3095.24 | 10643 |
| X6B | 714.286 | 10000 |
| X7B | 5000 | 0 |
| X8B | 4285.71 | 0 |
| Total Cost | $267,200 | $816,000\* |
| Waste Cost for Extra Pieces | $5692.9 | $5692.9 |

\* For the total cost of the GP section, we use the similar formula like what we did in the LP section, but we delete the cost of unused pieces of 51 and 24.5 inches, since based on our goal in GP, these two kinds of pieces are desirable. We want as much as we can produce. So we think they should not be considered as unused waste.

***Table 6.2. LP and GP results for Type B Rod***

The above tables list the optimal solution for LP and GP for Rod Type B. For the LP section, the total minimum cost is $267,200. For the GP section, the total minimum cost is $816,000. Both LP and GP sections have waste cost for extra pieces of $5692.9

In the LP solution, we have rods of patterns 1,7 and 8 as well along with 4,5 and 6 but do not have patterns 2 and 3. However, the GP solution includes only patterns 2 through 6 but in much higher quantities in comparison to LP. The total number of type B rails to be purchased according to GP is 60000 whereas the required rails given by LP is only 23750.

The difference in the number of rails required and hence the associated high cost as given by the GP solution can be associated with the selected goals. As explained in **Appendix F**, the goals of minimising waste cost and meeting the requirements of 51 inch and 24.5 inch rods does not ensure minimizing the total cost which includes the cost of buying the raw material and the cost associated with removal of unused pieces. Refer to **Appendix K** for further details.

Hence the LP method will provide a smaller total minimum cost for both type A and type B rods.

# Dynamic Programming (DP) Model

## 1.0 Introduction

The final method that will be used to attempt to optimize the production plan by minimizing the overall cost is a method that comes from dynamic programming. While this method is still a form of linear programming, it involves using two models: one which is the generic model of minimizing production costs similar to the LP model mentioned above and the other is a secondary LP model with the objective of minimizing the reduced cost of the cutting patterns used. In other words, the secondary model’s goal is to optimize the cutting patterns where the variables are the vector of cutting patterns used in the primary model. Once there are no more patterns that minimize the reduced cost in the secondary model, the primary model will provide the optimal solution using the results of the secondary model. The secondary model is very similar to a commonly used model in the operations research field called the Knapsack model. The two models together should theoretically provide the optimal cutting patterns as well as the optimal production plan and cost.

## 2.0 Input Information

The input information for the dynamic model is similar to the classical linear programming model mentioned in previous sections. For this model to work, an initial set of cutting patterns must be provided. There are many methods that can be used to determine the initial set of patterns, however for simplification purposes, we used the same patterns from the LP and ILP sections of this report. Please refer to the LP section on input information.

## 3.0 Output Information for Dynamic Model

Since the dynamic programming model is optimizing over the patterns and the cost we would intuitively expect this model to have the best overall performance. However, in actuality, we see that the model performs only slightly better or similar to the LP/ILP models. This could be in part due to the physical limitations provided by the production team that were implemented as constraints such as only 5 cuts allowed or 8 cutting patterns maximum. As a result of these constraints, it seems that given the initial cutting patterns determined using the logic discussed in the previous section, we arrive at a basis in the subproblem where there are no additional vector variables with a negative reduced cost which could enter the basis. As such, the subproblem only runs a couple of iterations before maxing out. Although the results from this model did not necessarily prove to be the best in our case compared to the previous LP and ILP models, we would recommend further analysis to use different logic in determining the starting cutting patterns. Having a different initial pattern to start could very likely result in a different solution since the candidate vectors that would enter the basis would be different.

The tables in the section below show the cutting pattern generated using the dynamic programming method and the results of the master problem.

### 3.1 Output of Dynamic Model for Rod Type A

While the dynamic model provides non-integer solutions, it is clear that the model cutting patterns from previous sections (LP and ILP) are not the most optimal patterns since the dynamic programming model brought other patterns into the basis. Although in the final model there are 13 total patterns where we have a limitation of 8 possible patterns, not all 13 patterns are used and therefore we know will assume this model is still valid using these patterns. The best LP cost for rod type A was $346,771.43 and using the production plan listed in the table below, the total cost is $262,043.00 which outperformed the best model for rod type B. However, since the production plan is not in integer values, the model needs to be adapted to provide a plan usable for production. Please see section 4.0 of the dynamic programming module for more information.

|  | | # of pieces cut per pattern | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | Qty. | 22.5 | 24.5 | 49 | 50 | 51 | 55 |
| 1 | 357.143 | 3 | 0 | 2 | 0 | 0 | 0 |
| 2 | 1071.43 | 0 | 3 | 0 | 0 | 0 | 2 |
| 3 | 1164.29 | 1 | 0 | 0 | 1 | 0 | 2 |
| 4 | 0 | 2 | 1 | 0 | 0 | 0 | 2 |
| 5 | 0 | 1 | 2 | 0 | 0 | 0 | 2 |
| 6 | 5535.71 | 0 | 1 | 1 | 1 | 1 | 0 |
| 7 | 4464.29 | 1 | 0 | 0 | 0 | 1 | 2 |
| 8 | 0 | 0 | 1 | 0 | 0 | 2 | 1 |
| 9 | 11250 | 0 | 1 | 3 | 0 | 0 | 0 |
| 10 | 0 | 0 | 1 | 3 | 0 | 0 | 0 |
| 11 | 0 | 0 | 1 | 3 | 0 | 0 | 0 |
| 12 | 0 | 0 | 1 | 3 | 0 | 0 | 0 |
| 13 | 0 | 0 | 1 | 3 | 0 | 0 | 0 |

***Table 3.1.DP Output of Dynamic Model - Rod A***

### 3.2 Output of Dynamic Model for Rod Type B

Using the same initial patterns as the LP and ILP model (section 2.0 in LP module), the results of the dynamic programming model for rod Type B exactly match those of the LP. Therefore in selecting the best model for Rod Type B, the ILP model still remains the best model.

## 4.0 Conclusions of the Dynamic Model

The dynamic programming model in theory should have been the best performing model of all of the ones discussed in this paper. However, for Rod Type B, the model performance was exactly the same as the LP model. The interesting piece of information gained from the dynamic model is that the cutting patterns for rod type A are not necessarily optimal. Since the output of the model was not in integer and we would like to provide the production team with the exact number of rods needed with a corresponding production plan, we will extend the analysis of the dynamic model by running the ILP model once more for Rod Type A. The input information can be found in section 3.1 of the dynamic programming module of this report. To make this adjustment, we will swap cutting pattern 8 and 9 from the dynamic model into the original ILP model for rod type A. The change is reflected in the below table:

| Model - Rod Type A | Pattern i | # of 51 in. parts | # of 24.5 in. parts | # of 55 in. parts | # of 50 in. parts | # of 49 in. parts | # of 22.5 in. parts |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Original ILP | 8 | 2 | 1 | 1 | 0 | 0 | 0 |
| New ILP | 8 | 0 | 1 | 0 | 0 | 3 | 0 |

***Table 4.0.DP - Changes to Pattern 8 for ILP Model***

As a result of the changes to pattern 8 of the ILP model, the total cost from the original ILP model for Rod A goes from $346,771.43 to $262,057.92 -- a total of $84,713.51 in savings. As a result, the new ILP model was the overall best performing model analyzed. See the AMPL model formulation and output of the new ILP model in Appendix I and the subsequent executive report module for more information.

# 

# Report to Management

## 1.0 Executive Summary

Based on forecasted demand for our products, the supply chain team predicts that demand will increase considerably within the next quarter. Mr. Avila, the West Coast regional production manager, has determined that the Los Angeles plant is able to provide the capacity to meet demand and will begin ramping up production soon. Despite being able to meet the demand in the LA plant, his team estimates that 5000 additional material handling carts will be needed to transport our products throughout the plant and act as temporary storage racks while not in work. Mr. Avila has been working in close collaboration with several teams to design a production plan to produce the 5000 material handling units. The objective of this initiative was to be able to produce all required material handling units while minimizing the total cost for the company.

Our special projects team has been tasked with developing a production plan using mathematical optimization. With two suppliers, supplier A who could provide a 184 inch rod at $9.50 each or supplier B who could supply a 214 inch rod at $11.00 each, we needed to determine which supplier to contract (see project overview). The supply chain team’s negotiations required that we contract only a single supplier in order to lock in those prices. Based on our analysis, we have arrived at a total production cost of $262,057.92 using the supplier for Rod Type A. This total cost would include all raw materials, waste, and labor costs which are factored into the unit cost of each piece of raw material. Using this production plan, we can meet Mr. Avila’s plan to produce the 5000 metal baskets and ensure that we are minimizing the total cost of unused or wasted materials. The production plan to achieve the lowest possible cost is listed in the table below.

## 2.0 Analysis

### 2.1 Linear Programming

To come up with the above solution, we began by setting up and solving the most basic version of the model. To do this, a total of 8 cutting patterns were chosen with care to ensure that no additional components were able to be cut from any given rod while ensuring all of the limitations (Appendix A) provided by the production team were met. The initial patterns can be found in Linear Programming section 2.0 for both rods A and B. This analysis enabled our team to get a baseline production cost for each rod type that could be used for comparison purposes and led to the following total costs for each type of rod.

| Rod | Baseline Model |
| --- | --- |
| Type A | $346,771.43 |
| Type B | $267,200.00 |

***Table 2.0. Cost of baseline Model***

From the results of our initial model we were inclined to choose the supplier who could provide rod type B because it would result in the lowest production cost for our company. One downside to this initial baseline model was that it entailed purchasing fractional components of rods which is not feasible given the limitations of our contract. Unfortunately due to the nature of the model, it proved difficult to perform a what-if analysis because changing one small component could result in large changes to the model due to the interrelationships in the cutting patterns. One component of what-if analysis we are able to understand without changing the model entirely is that if the number of rods were to decrease, the overall production cost would decrease as well. See section 5.0 in the linear programming module.

### 2.2 Integer Linear Programming

Since the major downside of our baseline model was that it determined a production plan using fractional components, we wanted to ensure that we could provide the procurement team with the exact number of rods needed. To do this we built on top of our baseline model to ensure that only whole rods are included in the plan. As expected when using a full rod versus a fraction rod, the cost increases albeit very slightly. The costs of this new model are shown below in comparison to the baseline.

| Rod | Baseline Model | Full Rod Model |
| --- | --- | --- |
| Type A | $346,771.43 | $346,778.96 |
| Type B | $267,200.00 | $267,214.26 |

***Table 2.2. Cost of baseline Model and full rod model***

With the results being consistent with the baseline model but slightly higher, the team was still inclined to choose supplier B to contract with as it results in the lowest total cost.

### 2.3 Goal Programming

While we have met our goal of minimizing total cost and providing the factory with a plan for production, we wanted to determine what would happen if we were to develop goals for each model to see if this added flexibility could help to lower the cost of production.

We used three goals for each model where the first goal was to not exceed the production cost attributed to material waste from the baseline model, and other two goals were to manufacture at least a certain number of 51 inch parts and 24.5 inch parts to meet the demand respectively. The order and priority of these goals was discussed and agreed upon with a cross functional team within the LA Plant.

Upon further analysis while running the model using these goals, we were not able to successfully improve the total production cost. At this point in our analysis, we are still inclined to select supplier B and the production plan provided by the Full Rod (integer linear programming) Model from the section above.

### 2.4 Dynamic Programming Model

In an attempt to further optimize the production cost, the team developed a new and more advanced method which would optimize the cost but also optimize the cutting patterns used in the production plan. Rather than a static cutting plan, this plan would dynamically change based on what the model could use to lower the production cost even more.

After developing and running the model for both Rod Types, we determined that Rod B could not be optimized further by modifying the cutting patterns since our results matched those of the baseline model. However, by running the model for Rod Type A, we learned that at least one of our cutting patterns needed to change since the model introduced additional patterns. With this new pattern, the model performed better than any model we had previously examined with a total cost of $262,043.00. This enabled our team to solve the full rod model again with the new pattern to achieve a production plan without fractional rod components and a lower overall cost.

## 3.0 Conclusion & Recommendations

The small enhancement made to the cutting patterns of rod type A determined through the dynamic model enabled us to develop a full-scale production plan that:

* Meets the demand requirements to produce 5000 material handling units for the LA Plant
* Select a raw material supplier
* Provide a plan for production with exact lengths and amounts
* Minimize the total production cost

As mentioned in the Executive Summary portion of this module, we recommend that supplier A be chosen with the following production plan.

|  | | # of pieces cut per pattern | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Pattern i | Qty | 22.5 | 24.5 | 49 | 50 | 51 | 55 |
| 1 | 357 | 3 | 0 | 2 | 0 | 0 | 0 |
| 2 | 1072 | 0 | 3 | 0 | 0 | 0 | 2 |
| 3 | 4464 | 1 | 0 | 0 | 1 | 0 | 2 |
| 6 | 5536 | 0 | 1 | 1 | 1 | 1 | 0 |
| 7 | 4465 | 1 | 0 | 0 | 0 | 1 | 2 |
| 8 | 11250 | 0 | 1 | 0 | 0 | 3 | 0 |

***Table 3.0.1. Recommended Cutting method***

This plan would achieve the lowest total production cost of $262,057.92 which is $5,156.34 less than that lowest production plan using rod type B. The table below shows the comparisons between the lowest cost models for rod type A and B.

|  | **Rod A** | **Rod B** |
| --- | --- | --- |
| **Cost per unit** | $9.50 | $11.00 |
| **Stock length** | 184 in. | 214 in. |
| **Waste cost per inch** | $0.02 | $0.02 |
| **Total Waste Cost (scrap + unused)** | $4,189.92 | $5,954.28 |
| **Total Production Cost** | $262,057.92 | $267,214.26 |

***Table 3.0.2. Information of Each Type***

While we recommend selecting rod type A along with the production plan mentioned above, due to the overall lower cost, there might be some other implicit factors in the decision making process such as supplier quality, customer service, supplier relations, or lead time that should be considered. Some additional factors from a manufacturing and transportation perspective are:

* Since rod A is 30 inches shorter than rod B, it might be easier to store raw material by consuming less storage space
* Rod A would also be lighter and smaller and easier to transport which could lead to savings in transportation or labor costs
* The production plan for rod A uses 6 of 8 cutting patterns while the production plan for rod B uses 7 of 8 cutting patterns, using rod A, could save time in programming the cutting machine as well as save changeover time when switching to a different cutting pattern (see LP section 2.2 for rod B production plan)

Since the difference in cost between rods A and B is nominal in relation to the overall cost, either supplier or production plan would provide similar results and therefore either production plan is a viable option if for some reason or limitation, the company would need to select rod B. Otherwise, selecting the production plan using rod A is highly recommended.

# 

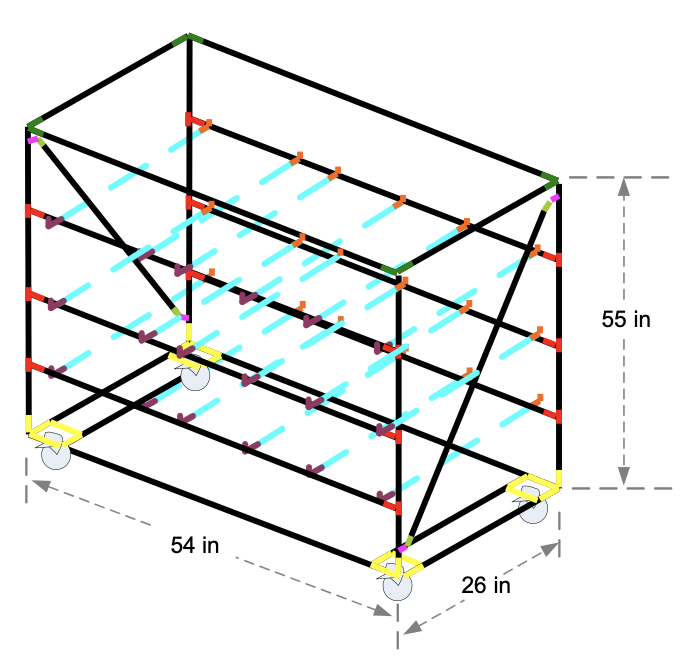
# 

# Appendices

## Appendix A: Original Problem, Assumptions, Background Information

### Appendix A.1 Problem Statement

**We need to produce at least 5000 baskets** to be used for moving boxes around a factory. The related information and Bill of Material (BOM) are below (provided by Mr. Avila). The required pieces of bars are cut from metal rails.



***Figure A.1: Engineering Drawing***

| Part | Base | | Sides | | Levels | Upper Frame | |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Piece | Length | Width | Height | Supports | Length | Length | Width |
| Qty. | 2 | 4 | 4 | 2 | 6 | 2 | 2 |
| Inches | 51 | 24.5 | 55 | 50 | 49 | 49 | 22.5 |

***Table A.1: Bill of Materials (BoM)***

There are two types of metal rails. Type A metal rails each with length of 184 inches and cost $9.50. Type B metal rails each have a length of 214 inches and cost $11.00. There is a cost associated with removal of the waste and unused pieces of the rails for about 2 cents per inch.

There are some limitations on the cutting process of these metal rails.

* **Limitation 1:** Each metal rail can be cut at most 5 times (into 6 pieces, whether being used or waste).
* **Limitation 2:** At most 8 different patterns can be planned for cutting.
* **Limitation 3:** Only one type of metal rails can be selected for all the cuts.

We would like to minimize the total cost. Which type of metal rails should be purchased? How many? What will be the cutting plan (patterns)?

### Appendix A.2 Modeling Assumptions

* Assume that the company has unlimited resources, like stopless machines, workers working 24 hours per day, 7 days per week, meaning that they could produce as many rods as they want. All we want to do is to finish the 5000 brackets and minimize the total costs. In addition, under this assumption it is assumed that the cost of each rod includes labor for regular time and overtime needed to produce the metal baskets.
* Rods will be regarded as waste when calculating the total cost but could be used later.
* The motivation of choosing goal 1 is that we want to decrease our total cost, and the total waste cost from extra pieces wasted is part of our total cost. To decrease total waste cost from extra pieces wasted can help us decrease our total cost.
* We assume that the production department prefers to produce 51 inch pieces and 24.5 inch pieces. The market demand for 51 inch pieces and 24.5 inch pieces is really high since almost all sizes of brackets require these two kinds of pieces. With the pandemic passing by and the recovery of the economy, domestic demand and export are increasing sharply, putting 51 inch pieces and 24.5 inch pieces into a vital position.
* For sensitivity analysis, assume that for any type of part the quantity per unit can increase or decrease by at most 2 pieces.

### Appendix A.3 Problem Source

This problem was provided by Professor [Sima Parisay](mailto:parisay@usc.edu), PhD who has received the detailed requirements including bill of materials and engineering drawings from the factory manager Mr. Avila.

### Appendix A.4 Software Used

The software used to solve the linear programming model was: **AMPL IDE Version: 3.6.7.202106142233** with CPLEX solver. The following commands can be used to run the model from the AMPL console:

ampl: reset;

ampl: option solver cplex;

ampl: model modelname.mod;

ampl: solve;

Note: “modelname.mod” is an arbitrary name. Please refer to the sections A.2.1 and A.2.2 and save those models as a .mod file before running the above console commands.

## Appendix B: Detailed Explanation of Problem and Formulation for LP

As for type A:

**minimize** z: 9.5 \* (X1A + X2A + X3A + X4A + X5A + X6A + X7A + X8A) + 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A + 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)

+ 0.02 \* (

((0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A) - 2 \* 5000)\* 51.0 +

((0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A) - 4 \* 5000)\* 24.5 +

((0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A) - 4 \* 5000)\* 55.0 +

((0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 2 \* 5000)\* 50.0 +

((2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 8 \* 5000)\* 49.0 +

((3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A) - 2 \* 5000)\* 22.5 );

Constraints:

C51: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A ) >= 2 \* 5000;

C24: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) >= 4 \* 5000;

C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

X1A, X2A, X3A, X4A, X5A, X6A, X7A, X8A >=0;

We listed all the possible patterns which are more than 70. By selecting those 8 with least leftovers, we hope we are able to minimize the total cost. Pattern 1 with 18.5 unused materials, pattern 2 with 0.5 unused materials, pattern 3 with 1.5 unused materials, pattern 4 with 4.5 unused materials, pattern 5 with 2.5 unused materials, pattern 6 with 9.5 unused materials, pattern 7 with 0.5 unused materials and pattern with 2.5 unused materials.

As for type B:

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**minimize** z: 11.0 \* (X1B + X2B + X3B + X4B + X5B + X6B + X7B + X8B)

+ 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B)

+ 0.02 \* (

((0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B) - 2 \* 5000)\* 51.0 +

((0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B) - 4 \* 5000)\* 24.5 +

((0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B) - 4 \* 5000)\* 55.0 +

((0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B) - 2 \* 5000)\* 50.0 +

((4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B) - 8 \* 5000)\* 49.0 +

((0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B) - 2 \* 5000)\* 22.5 );

Constraints:

**subject** **to** C51: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

X1B, X2B, X3B, X4B, X5B, X6B, X7B, X8B >=0;

We listed all the possible patterns which are more than 90. By selecting those with least leftovers and changing different patterns, we hope we are able to minimize the total cost and get the non-integer solution. And luckily Daniel found a combination with no waste rob.

Pattern 1 with 18 unused materials, pattern 2 with 15 unused materials, pattern 3 with 0 unused materials, pattern 4 with 2 unused materials, pattern 5 with 15 unused materials, pattern 6 with 3 unused materials, pattern 7 with 4 unused materials and pattern with 14 unused materials.

### Appendix B.2 Detailed Explanation of Formulation

To formulate the problem described in the project overview, it is necessary to determine the information that the company needs in order to begin production. This information includes:

* Which rod A or B should be used to produce the 5,000 metal baskets?
* How many rods of the selected type will be needed to meet my demand?
* How much will it cost to manufacture the products with the least amount of waste?
* Which cutting patterns will be used in the manufacturing process?

After determining the information that must be provided to the company, the team decided that in order to reduce complexity, two models would be created: a model for rod type A and a model for rod type B. Since each model would have the same objective of minimizing the total production cost, the outputs of each model could be compared to determine which type of rod to select. In each model, 8 decision variables were used to reflect the limitation of the 8 possible cutting patterns. Each cutting pattern used also had a maximum of 5 cuts or 6 total pieces.

The cutting patterns for each rod type are listed in Section 2.0 Tables 2.2.1 and 2.1.2. These patterns were selected by ensuring that all possible pieces that can be cut from the bar are cut. Put simply, given the remainder of scrap material, no other piece of a valid length could be cut from it. Additionally, the patterns were selected so that the amount of scrap material in inches is as low as possible while ensuring that the combination of all cut patterns includes every possible piece necessary. The cutting patterns are defined as the decision variables in each model where XiA, i = 1,...,8 is the cutting pattern *i* using rod type B and XiB, i = 1,...,8 is cutting pattern *i* using rod type B. It is important to note that the models for type A and B are two separate models; however, the A and B distinction on the decision variables is to simplify communication. Additionally, X1A and X1B will not necessarily have the same pattern (i.e., number of each part cut using pattern 1). For example, X1A could have 3 51” pieces and 2 49” pieces and X1B may have 1 55” piece and 4 24.5” pieces (for the actual cutting patterns please refer to the formulation in the AMPL program (section A.2) or the summary tables in section 2.0).

The next component of the formulation is to determine the objective function. With the overall goal of minimizing production costs and meeting demand, the objective function formulation has three major components. Written simply this is:

**Min.** Total Cost = (Cost of all rods used) + (Scrap material cost) + (Unused piece cost)

The different components of the objective function can be further decomposed into:

(Cost of all rods used) = CZ\*(X1Z+X2Z+X3Z+X4Z+X5Z+X6Z+X7Z+X8Z)

(Scrap material cost) = S\*(w1\*X1Z+w2\*X2Z+w3\*X3Z+w4\*X4Z +

w5\*X5Z+w6\*X6Z+w7\*X7Z+w8\*X8Z)

(Unused piece cost) = S\*(

((N51) - 2 \* 5000)\*51 +

((N24) - 4 \* 5000)\*24.5 +

((N55) - 4 \* 5000)\*55 +

((N50) - 2 \* 5000)\*50 +

((N49) - 8 \* 5000)\*49 +

((N22) - 2 \* 5000)\*22.5)

CZ = cost of rod Z; Z = A,B; S = cost of scrap material; wi = scrap waste for pattern *i*;

*i* = 1,...,8; NJ = number of pieces of type j produced

Lastly the final component of the formulation includes setting the constraints. Given the method of formulation which resulted in two individual models, the careful way in which the cutting patterns were chosen to not exceed 8 total patterns, and the choice of cutting patterns with a maximum of 5 cuts, the only constraints in either model are the limitations on demand. This allows the constraints to be written simply as shown below:

*Subject to:*

∑ Pi,j,Z \* XiZ ≥ Dj

Where: *i* is cutting pattern, i = 1,...,8

*j* is part of type j = 50, 49, 51, 55, 24.5, 22.5

Z is rod type Z = A,B

Dj is demand associated with part type j

## Appendix C: AMPL for LP Model

### Appendix C.1 Rod Type A LP Model - AMPL Input

# LINEAR PROGRAMMING MODEL (LP)

## Given:

# ROD TYPE: A

# ROD LENGTH: 184 inches

# ROD COST: $9.50 per rod

# COST OF WASTE/UNUSED: $0.02 per inch

# define decision variables XiA, i = 1,...,8 is cutting pattern i cut using rod A

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**minimize** z: 9.5 \* (X1A + X2A + X3A + X4A + X5A + X6A + X7A + X8A) # cost of type A rods used $9.5 per rod

+ 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A + 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A) # cost of wasted material from each cut

+ 0.02 \* (

((0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A) - 2 \* 5000)\* 51.0 + # cost of unused pieces of 51 inches

((0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A) - 4 \* 5000)\* 24.5 + # cost of unused pieces of 24.5 inches

((0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A) - 4 \* 5000)\* 55.0 + # cost of unused pieces of 55 inches

((0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 2 \* 5000)\* 50.0 + # cost of unused pieces of 50 inches

((2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 8 \* 5000)\* 49.0 + # cost of unused pieces of 49 inches

((3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A) - 2 \* 5000)\* 22.5 ); # cost of unused pieces of 22.5 inches

#constraints for demand of X inch parts

**subject** **to** C51: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

### 

### Appendix C.2 Rod Type B LP Model - AMPL Input

# LINEAR PROGRAMMING MODEL (LP)

## Given:

# ROD TYPE: B

# ROD LENGTH: 214 inches

# ROD COST: $11.00 per rod

# COST OF WBSTE/UNUSED: $0.02 per inch

# define decision variables XiB, i = 1,...,8 is cutting pattern i cut using rod B

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**minimize** z: 11.0 \* (X1B + X2B + X3B + X4B + X5B + X6B + X7B + X8B) # cost of type B rods used $11.0 per rod

+ 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) # cost of wasted material from each cut

+ 0.02 \* (

((0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B) - 2 \* 5000)\* 51.0 + # cost of unused pieces of 51.0 inches

((0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B) - 4 \* 5000)\* 24.5 + # cost of unused pieces of 24.5 inches

((0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B) - 4 \* 5000)\* 55.0 + # cost of unused pieces of 55.0 inches

((0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B) - 2 \* 5000)\* 50.0 + # cost of unused pieces of 50.0 inches

((4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B) - 8 \* 5000)\* 49.0 + # cost of unused pieces of 49.0 inches

((0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B) - 2 \* 5000)\* 22.5 ); # cost of unused pieces of 22.5 inches

# constraints for demand of X inch parts

**subject** **to** C51: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

### Appendix C.3 AMPL Output for Rod Type A LP Model

ampl: reset;

option solver cplex;

model finalreport\_LP\_RodA.mod;

solve;

display X1A, X2A, X3A, X4A, X5A, X6A, X7A, X8A;

CPLEX 20.1.0.0: optimal solution; objective 346771.4286

6 dual simplex iterations (0 in phase I)

ampl:

X1A = 16428.6

X2A = 4285.71

X3A = 2857.14

X4A = 0

X5A = 0

X6A = 7142.86

X7A = 2857.14

X8A = 0

* 16428.6 pattern 1 will be produced and the purchase cost is $156071.7. We can get 32857.2 robs with 49 inches and 49285.8 robs with 22.5 inches. The total waste is 303929.1 and costs $6078.58.
* 4285.71 pattern 2 will be produced and the purchase cost is $40714.25. We can get 12857.13 robs with 24.5 inches and 8571.42 robs with 55 inches. The total waste is 2142.85 and costs $42.85.
* 2857.14 pattern 3 will be produced and the purchase cost is $27142.83. We can get 5714.28 robs with 55 inches and 2857.14 robs with 50 inches and 2857.14 robs with 22.5 inches. The total waste is 4285.71 and costs $85.7142.
* 7142.86 pattern 6 will be produced and the purchase cost is $67857.17. We can get 7142.86 robs with 24.5 inches, 7142.86 robs with 24.5 inches, 7142.86 robs with 50 inches and 7142.86 robs with 49 inches. The total waste is 67857.17 and costs $1357.14.
* 2857.14 pattern 7 will be produced and the purchase cost is $27142.83. We can get 2857.14 robs with 51 inches, 5714.28 robs with 55 inches and 2857.14 robs with 22.5 inches. The total waste is 1428.57 and costs $28.57.

### 

### Appendix C.4 AMPL Output for Rod Type B LP Model

reset;

option solver cplex;

model finalreport\_LP\_RodB.mod;

solve;

display X1B, X2B, X3B, X4B, X5B, X6B, X7B, X8B;

CPLEX 20.1.0.0: optimal solution; objective 267200

7 dual simplex iterations (0 in phase I)

X1B = 9226.19

X2B = 0

X3B = 0

X4B = 1428.57

X5B = 3095.24

X6B = 714.286

X7B = 5000

X8B = 4285.71

## 

* 9226.19 pattern 1 will be produced and the purchase cost is $101488.09. We can get 36904.76 rods with 49 inches. The total waste is 166071.42 and costs $3321.4284.
* 1428.57 pattern 4 will be produced and the purchase cost is $15714.27. We can get 1428.57 robs with 24.5 inches, 4285.71 robs with 55 inches and 1428.571 robs with 22.5 inches. The total waste is 2857.14 and costs $57.1428.
* 3095.24 pattern 5 will be produced and the purchase cost is $34047.64. We can get 9285.72 robs with 50 inches and 3095.24 robs with 49 inches. The total waste is 46428.6 and costs $928.572.
* 714.83 pattern 6 will be produced and the purchase cost is $7857.146. We can get 714.286 robs with 51 inches, 1428.572 robs with 55 inches, 714.286 robs with 50 inches. The total waste is 15000 and costs $300.
* 5000 pattern 7 will be produced and the purchase cost is $55000. We can get 5000 robs with 51 inches, 10000 robs with 24.5 inches and 10000 robs with 55 inches. The total waste is 20000 and costs $400.
* 4285.71 pattern 8 will be produced and the purchase cost is $47142.81. We can get 4285.71 robs with 51 inches, 8571.42 robs with 24.5 inches, 8571.42 robs with 22.5 inches and4285.71 robs with 55 inches. The total waste is 34285.68 and costs $685.7136.

## 

## Appendix D: LP Sensitivity Analysis

### Appendix D.1 Sensitivity Analysis for Rod A LP

ampl: reset;

option solver cplex;

option cplex\_options sensitivity; # indicate sensitivity module

model finalreport\_LP\_RodA.mod;

solve;

display X1A, X2A, X3A, X4A, X5A, X6A, X7A, X8A;

### Sensitivity Analysis: objective function coeff

display \_varname,\_var,\_var.rc, \_var.down, \_var.current, \_var.up;

### Sensitivity Analysis: constraints

display \_conname,\_con.slack,\_con.down, \_con.current, \_con.up, \_con.dual;

CPLEX 20.1.0.0: sensitivity

CPLEX 20.1.0.0: optimal solution; objective 346771.4286

6 dual simplex iterations (0 in phase I)

suffix up OUT;

suffix down OUT;

suffix current OUT;

X1A = 16428.6

X2A = 4285.71

X3A = 2857.14

X4A = 0

X5A = 0

X6A = 7142.86

X7A = 2857.14

X8A = 0

ampl: : \_varname \_var \_var.rc \_var.down \_var.current \_var.up :=

1 X1A 16428.6 0 9.58545 13.18 26.36

2 X2A 4285.71 0 9.885 13.18 14.498

3 X3A 2857.14 4.44089e-16 11.3827 13.18 19.77

4 X4A 0 1.88286 11.2971 13.18 1e+20

5 X5A 0 0.941429 12.2386 13.18 1e+20

6 X6A 7142.86 -1.77636e-15 6.59 13.18 14.9773

7 X7A 2857.14 0 8.2375 13.18 14.3429

8 X8A 0 1.41214 11.7679 13.18 1e+20

;

: \_conname \_con.slack \_con.down \_con.current \_con.up \_con.dual

:=

1 C51 -3.63798e-12 5000 10000 16666.7 2.82429

2 C24 0 5000 20000 40000 0.941429

3 C55 0 6666.67 20000 53333.3 5.17786

4 C50 0 5000 10000 16666.7 2.82429

5 C49 0 10000 40000 1e+20 6.59

6 C22 45000 -1e+20 10000 55000 0

;

### 

### Appendix D.2 Sensitivity Analysis for Rod B LP

ampl: reset;

option solver cplex;

option cplex\_options sensitivity; # indicate sensitivity module

model finalreport\_LP\_RodB.mod;

solve;

display X1B, X2B, X3B, X4B, X5B, X6B, X7B, X8B;

### Sensitivity Analysis: objective function coeff

display \_varname,\_var,\_var.rc, \_var.down, \_var.current, \_var.up;

### Sensitivity Analysis: constraints

display \_conname,\_con.slack,\_con.down, \_con.current, \_con.up, \_con.dual;

CPLEX 20.1.0.0: sensitivity

CPLEX 20.1.0.0: optimal solution; objective 267200

7 dual simplex iterations (0 in phase I)

suffix up OUT;

suffix down OUT;

suffix current OUT;

X1B = 9226.19

X2B = 0

X3B = 0

X4B = 1428.57

X5B = 3095.24

X6B = 714.286

X7B = 5000

X8B = 4285.71

: \_varname \_var \_var.rc \_var.down \_var.current \_var.up :=

1 X1B 9226.19 0 0 15.28 15.28

2 X2B 0 2.8 11.46 14.26 1e+20

3 X3B 0 0 15.28 15.28 1e+20

4 X4B 1428.57 0 1.91 15.28 15.28

5 X5B 3095.24 0 15.28 15.28 19.48

6 X6B 714.286 0 12.48 15.28 15.28

7 X7B 5000 0 11.46 15.28 18.08

8 X8B 4285.71 0 15.28 15.28 42.02

;

: \_conname \_con.slack \_con.down \_con.current \_con.up \_con.dual :=

1 C51 0 9000 10000 12500 3.82

2 C24 0 10000 20000 21428.6 1.91

3 C55 0 15000 20000 50000 3.82

4 C50 -1.81899e-12 714.286 10000 120714 3.82

5 C49 0 3095.24 40000 1e+20 3.82

6 C22 0 0 10000 20000 1.91

;

### Appendix D.3 Constraint Sensitivity Analysis for Rod B Calculations

#### Decreases to 49 inch part

Z' = Z + ∆Z1

∆Z1 = (30000-40000) \* (3.82) = -38200

Z' = 267200 + -38200 = 229,000

Based on the calculation above we can see that by decreasing the number of 49 inch parts from 8 to 6 changes the overall demand from 40,000 pieces to 30,000 pieces. As a result of this change, the objective function would decrease from $267,200.00 to $229,000. This reflects a $38,200 improvement to Z, the objective function.

#### Increases to 49 inch part

Z' = Z + ∆Z1

∆Z1 = (50000-40000) \* (3.82) = 38200

Z' = 267200 + 38200 = 305,400

Based on the calculation above we can see that by increasing the number of 49 inch parts from 8 to 10 changes the overall demand from 40,000 pieces to 50,000 pieces. As a result of this change, the objective function would increase from $267,200.00 to $305,400. This reflects a $38,200 loss to Z, the objective function.

#### Note on sensitivity analysis

Basic rules of thumbs indicate that for binding constraints, the most positive and the most negative shadow price constraints should be examined. The shadow price for the 49 inch piece was the most positive shadow price. Since there are no negative shadow prices and all constraints are binding, we will not proceed further with sensitivity analysis for the LP model. Especially given the interrelationship of the variables and constraints where if the overall number of baskets changes from 5000, it would be advised to solve the model again.

## 

## Appendix E: AMPL for ILP Model

### Appendix E.1 Rod Type A ILP Model - AMPL Input

# INTEGER LINEAR PROGRAMMING MODEL (ILP)

## Given:

# ROD TYPE: A

# ROD LENGTH: 184 inches

# ROD COST: $9.50 per rod

# COST OF WASTE/UNUSED: $0.02 per inch

# define decision variables XiA, i = 1,...,8 is cutting pattern i cut using rod A

**var** X1A **integer** >=0;

**var** X2A **integer** >=0;

**var** X3A **integer** >=0;

**var** X4A **integer** >=0;

**var** X5A **integer** >=0;

**var** X6A **integer** >=0;

**var** X7A **integer** >=0;

**var** X8A **integer** >=0;

**minimize** z: 9.5 \* (X1A + X2A + X3A + X4A + X5A + X6A + X7A + X8A) # cost of type A rods used $9.5 per rod

+ 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A + 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A) # cost of wasted material from each cut

+ 0.02 \* (

((0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A) - 2 \* 5000)\* 51.0 + # cost of unused pieces of 51 inches

((0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A) - 4 \* 5000)\* 24.5 + # cost of unused pieces of 24.5 inches

((0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A) - 4 \* 5000)\* 55.0 + # cost of unused pieces of 55 inches

((0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 2 \* 5000)\* 50.0 + # cost of unused pieces of 50 inches

((2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 8 \* 5000)\* 49.0 + # cost of unused pieces of 49 inches

((3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A) - 2 \* 5000)\* 22.5 ); # cost of unused pieces of 22.5 inches

# constraints for demand of X inch parts

**subject** **to** C51: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

### 

### Appendix E.2 Rod Type B ILP Model - AMPL Input

# INTEGER LINEAR PROGRAMMING MODEL (ILP)

## Given:

# ROD TYPE: B

# ROD LENGTH: 214 inches

# ROD COST: $11.00 per rod

# COST OF WBSTE/UNUSED: $0.02 per inch

# define decision variables XiB, i = 1,...,8 is cutting pattern i cut using rod B

**var** X1B **integer** >=0;

**var** X2B **integer** >=0;

**var** X3B **integer** >=0;

**var** X4B **integer** >=0;

**var** X5B **integer** >=0;

**var** X6B **integer** >=0;

**var** X7B **integer** >=0;

**var** X8B **integer** >=0;

**minimize** z: 11.0 \* (X1B + X2B + X3B + X4B + X5B + X6B + X7B + X8B) # cost of type B rods used $11.0 per rod

+ 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) # cost of wasted material from each cut

+ 0.02 \* (

((0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B) - 2 \* 5000)\* 51.0 + # cost of unused pieces of 51.0 inches

((0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B) - 4 \* 5000)\* 24.5 + # cost of unused pieces of 24.5 inches

((0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B) - 4 \* 5000)\* 55.0 + # cost of unused pieces of 55.0 inches

((0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B) - 2 \* 5000)\* 50.0 + # cost of unused pieces of 50.0 inches

((4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B) - 8 \* 5000)\* 49.0 + # cost of unused pieces of 49.0 inches

((0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B) - 2 \* 5000)\* 22.5 ); # cost of unused pieces of 22.5 inches

# constraints for demand of X inch parts

**subject** **to** C51: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

## 

## 

### Appendix E.3 AMPL Output for Rod Type A ILP Model

ampl: reset;

option solver cplex;

model finalreport\_ILP\_RodA.mod;

solve;

display X1A, X2A, X3A, X4A, X5A, X6A, X7A, X8A;

CPLEX 20.1.0.0: optimal integer solution; objective 346778.96

6 MIP simplex iterations

0 branch-and-bound nodes

X1A = 16429

X2A = 4286

X3A = 2857

X4A = 0

X5A = 0

X6A = 7143

X7A = 2857

X8A = 0

### Appendix E.3 AMPL Output for Rod Type B ILP Model

ampl: reset;

option solver cplex;

model finalreport\_ILP\_RodB.mod;

solve;

display X1B, X2B, X3B, X4B, X5B, X6B, X7B, X8B;

CPLEX 20.1.0.0: optimal integer solution within mipgap or absmipgap; objective 267214.26

7 MIP simplex iterations

0 branch-and-bound nodes

absmipgap = 14.26, relmipgap = 5.33654e-05

X1B = 9226

X2B = 1

X3B = 0

X4B = 1428

X5B = 3095

X6B = 715

X7B = 5000

X8B = 4286

## Appendix F: Details of GP Model Formulation

## Formulation for Rod type A GP Model:

Objective function(s):

Min Z1=P1S1P

Min Z2=P2S2N

Min Z3=P3S3N

## 

Constraints:

The following constraint are about achieving Goal 1 (G1) - waste cost from extra pieces wasted should be at most $7592.9:

0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- S1P + S1N =7592.9;

XiA refers to the number of bars used for pattern i. The term −𝑆1𝑃 +𝑆1𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are about achieving Goal 2 (G2) - The number of 51 inch pieces should be at least 10000:

1 \*X6A + 1 \*X7A + 2 \*X8A - S2P + S2N= 2 \* 5000;

XiA refers to the number of bars used for pattern i. The right hand side is equal to 2\*5000=10000, because for each metal basket we need two 51 inch pieces, and we are producing 5000 metal baskets. The term −𝑆2𝑃 +𝑆2𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are about achieving Goal 3 (G3) - The number of 24.5 inch pieces should be at least 20000:

3 \*X2A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 1 \*X8A - S3P + S3N= 4 \* 5000;

XiA refers to the number of bars used for pattern i. The right hand side is equal to 4\*5000=20000, because for each metal basket we need four 24.5 inch pieces, and we are producing 5000 metal baskets. The term −𝑆3𝑃 +𝑆3𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are unchanged from LP and ILP section:

2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 2 \*X7A + 1 \*X8A >= 4 \* 5000;

1 \*X3A + 1 \*X6A >= 2 \* 5000;

2 \*X1A + 1 \*X6A >= 8 \* 5000;

3 \*X1A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 1 \*X7A >= 2 \* 5000;

In addition, we assume XiA are non-negative.

## Formulation for Rod type B GP Model:

Objective function(s):

Min Z1=P1S1P

Min Z2=P2S2N

Min Z3=P3S3N

Constraints:

The following constraint are about achieving Goal 1 (G1) - waste cost from extra pieces wasted should be at most $5692.9:

0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - S1P + S1N = 5692.9;

XiB refers to the number of bars used for pattern i. The term −𝑆1𝑃 +𝑆1𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are about achieving Goal 2 (G2) - The number of 51 inch pieces should be at least 10000:

1 \*X6B + 1 \*X7B + 1 \*X8B - S2P + S2N = 2 \* 5000;

XiB refers to the number of bars used for pattern i. The right hand side is equal to 2\*5000=10000, because for each metal basket we need two 51 inch pieces, and we are producing 5000 metal baskets. The term −𝑆2𝑃 +𝑆2𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are about achieving Goal 3 (G3) - The number of 24.5 inch pieces should be at least 20000:

2 \*X2B + 1 \*X4B + 2 \*X7B + 2 \*X8B - S3P + S3N = 4 \* 5000;

XiB refers to the number of bars used for pattern i. The right hand side is equal to 4\*5000=20000, because for each metal basket we need four 24.5 inch pieces, and we are producing 5000 metal baskets. The term −𝑆3𝑃 +𝑆3𝑁 here is to ensure we can satisfy the equality constraint through deviational variables.

The following constraints are unchanged from LP and ILP section:

3 \*X3B + 3 \*X4B +2 \*X6B + 2 \*X7B + 1 \*X8B >= 4 \* 5000;

1 \*X2B + 3 \*X5B + 1 \*X6B >= 2 \* 5000;

4 \*X1B + 1 \*X2B + 1 \*X3B + 1 \*X5B >= 8 \* 5000;

1 \*X4B + 2 \*X8B >= 2 \* 5000;

In addition, we assume XiB are non-negative.

## 

## Appendix G1: AMPL for Type A GP Model

**# code for Type A goal 1**

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s1\_plus;

**subject** **to** G1: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s1\_plus + s1\_minus =7592.9;

**subject** **to** C51G2: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s2\_plus + s2\_minus= 2 \* 5000;

**subject** **to** C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**# code for Type A goal 2**

**var X1A >= 0;**

**var X2A >= 0;**

**var X3A >= 0;**

**var X4A >= 0;**

**var X5A >= 0;**

**var X6A >= 0;**

**var X7A >= 0;**

**var X8A >= 0;**

**var s1\_plus >= 0;**

**var s1\_minus >= 0;**

**var s2\_plus >= 0;**

**var s2\_minus >= 0;**

**var s3\_plus >= 0;**

**var s3\_minus >= 0;**

**minimize z: s2\_minus;**

**subject to G1: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s1\_plus + s1\_minus =7592.9;**

**subject to C51G2: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s2\_plus + s2\_minus= 2 \* 5000;**

**subject to C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;**

**subject to C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;**

**subject to C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;**

**subject to C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;**

**subject to C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;**

**subject to g1: s1\_plus = 0;**

**# code for Type A goal 3**

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_minus;

**subject** **to** G1: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s1\_plus + s1\_minus =7592.9;

**subject** **to** C51G2: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s2\_plus + s2\_minus= 2 \* 5000;

**subject** **to** C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** g1: s1\_plus = 0;

**subject** **to** g2: s2\_minus = 0;

## Appendix G2: AMPL output for Type A GP Model

#goal 1

ampl: reset;

ampl: option solver cplex;

ampl: model g1.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 1550

3 dual simplex iterations (0 in phase I)

ampl: reset;

ampl: option solver cplex;

ampl: model g1.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

2 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15000

X2A = 14290

X3A = 0

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 32870

s3\_minus = 0

#goal 2

ampl: reset;

ampl: model g2.mod;

ampl: option solver cplex;

ampl: expand;

minimize z:

s2\_minus;

subject to G1:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s1\_plus + s1\_minus = 7592.9;

subject to C51G2:

X6A + X7A + 2\*X8A - s2\_plus + s2\_minus = 10000;

subject to C245G3:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_plus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

6 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15000

X2A = 7855

X3A = 2145

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 13565

s3\_minus = 0

#goal3

ampl: reset;

ampl: option solver cplex;

ampl: model g3.mod;

ampl: expand;

minimize z:

s3\_minus;

subject to G1:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s1\_plus + s1\_minus = 7592.9;

subject to C51G2:

X6A + X7A + 2\*X8A - s2\_plus + s2\_minus = 10000;

subject to C245G3:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_plus = 0;

subject to g2:

s2\_minus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

5 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15572

X2A = 8570

X3A = 1144

X4A = 0

X5A = 0

X6A = 8856

X7A = 0

X8A = 572

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 15138

s3\_minus = 0

## 

## Appendix G3: SA for Type A GP Model

Number of 51 > Waste > Number of 24.5

Goal 1:

var X1A >= 0;

var X2A >= 0;

var X3A >= 0;

var X4A >= 0;

var X5A >= 0;

var X6A >= 0;

var X7A >= 0;

var X8A >= 0;

var s1\_plus >= 0;

var s1\_minus >= 0;

var s2\_plus >= 0;

var s2\_minus >= 0;

var s3\_plus >= 0;

var s3\_minus >= 0;

minimize z: s1\_minus;

subject to C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

subject to G2: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

subject to C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

subject to C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

subject to C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

subject to C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

subject to C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**Goal 2:**

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s2\_plus;

**subject** **to** C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

**subject** **to** G2: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

**subject** **to** C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

Goal 3:

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_minus;

**subject** **to** C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

**subject** **to** G2: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

**subject** **to** C245G3: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

**subject** **to** g2: s2\_plus = 0;

Goal 1:

ampl: reset;

ampl: option solver cplex;

ampl: model g1.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15115.9

X2A = 10000

X3A = 0

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 20000

s3\_minus = 0

Goal 2:

ampl: reset;

ampl: option solver cplex;

ampl: model g2.mod;

ampl: expand;

minimize z:

s2\_plus;

subject to C51G2:

X6A + X7A + 2\*X8A - s1\_plus + s1\_minus = 10000;

subject to G1:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s2\_plus + s2\_minus = 7592.9;

subject to C245G3:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_minus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

6 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15000

X2A = 7855

X3A = 2145

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 13565

s3\_minus = 0

Goal 3:

ampl: reset;

ampl: option solver cplex;

ampl: model g3.mod;

ampl: expand;

minimize z:

s3\_minus;

subject to C51G2:

X6A + X7A + 2\*X8A - s1\_plus + s1\_minus = 10000;

subject to G1:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s2\_plus + s2\_minus = 7592.9;

subject to C245G3:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_minus = 0;

subject to g2:

s2\_plus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

5 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15572

X2A = 8570

X3A = 1144

X4A = 0

X5A = 0

X6A = 8856

X7A = 0

X8A = 572

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 15138

s3\_minus = 0

Number of 51 > Number of 24.5 > Waste

**Goal 1:**

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s1\_minus;

**subject** **to** C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

**subject** **to** C245G2: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** G3: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**Goal 2:**

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s2\_minus;

**subject** **to** C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

**subject** **to** C245G2: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** G3: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

**Goal 3:**

**var** X1A >= 0;

**var** X2A >= 0;

**var** X3A >= 0;

**var** X4A >= 0;

**var** X5A >= 0;

**var** X6A >= 0;

**var** X7A >= 0;

**var** X8A >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_plus;

**subject** **to** C51G1: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 2 \*X8A )- s1\_plus + s1\_minus= 2 \* 5000;

**subject** **to** C245G2: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) - s3\_plus + s3\_minus= 4 \* 5000;

**subject** **to** G3: 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A+ 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 2.5\*X8A)- s2\_plus + s2\_minus =7592.9;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

**subject** **to** g2: s2\_plus = 0;

Goal 1:

ampl: reset;

ampl: option solver cplex;

ampl: model g1.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15115.9

X2A = 10000

X3A = 0

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 20000

s3\_minus = 0

Goal 2:

ampl: reset;

ampl: option solver cplex;

ampl: model g2.mod;

ampl: expand;

minimize z:

s2\_minus;

subject to C51G1:

X6A + X7A + 2\*X8A - s1\_plus + s1\_minus = 10000;

subject to C245G2:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to G3:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s2\_plus + s2\_minus = 7592.9;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_minus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 15115.9

X2A = 10000

X3A = 0

X4A = 0

X5A = 0

X6A = 10000

X7A = 0

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 20000

s3\_minus = 0

Goal 3:

ampl: reset;

ampl: option solver cplex;

ampl: model g3.mod;

ampl: expand;

minimize z:

s3\_plus;

subject to C51G1:

X6A + X7A + 2\*X8A - s1\_plus + s1\_minus = 10000;

subject to C245G2:

3\*X2A + X4A + 2\*X5A + X6A + X8A - s3\_plus + s3\_minus = 20000;

subject to G3:

0.37\*X1A + 0.01\*X2A + 0.03\*X3A + 0.09\*X4A + 0.05\*X5A + 0.19\*X6A +

0.01\*X7A + 0.05\*X8A - s2\_plus + s2\_minus = 7592.9;

subject to C55:

2\*X2A + 2\*X3A + 2\*X4A + 2\*X5A + 2\*X7A + X8A >= 20000;

subject to C50:

X3A + X6A >= 10000;

subject to C49:

2\*X1A + X6A >= 40000;

subject to C22:

3\*X1A + X3A + 2\*X4A + X5A + X7A >= 10000;

subject to g1:

s1\_minus = 0;

subject to g2:

s2\_plus = 0;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

8 dual simplex iterations (0 in phase I)

ampl: display X1A,X2A,X3A,X4A,X5A,X6A,X7A,X8A,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1A = 16427.9

X2A = 4285.26

X3A = 2858.95

X4A = 0

X5A = 0

X6A = 7144.21

X7A = 2855.79

X8A = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

## 

## Appendix G4: AMPL for Type B GP Model

**# code for Type B goal 1**

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s1\_plus;

**subject** **to** G1: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s1\_plus + s1\_minus = 5692.9;

**subject** **to** G2: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s2\_plus + s2\_minus = 2 \* 5000;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**# code for Type B goal 2**

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s2\_minus;

**subject** **to** G1: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s1\_plus + s1\_minus = 5692.9;

**subject** **to** G2: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s2\_plus + s2\_minus = 2 \* 5000;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_plus = 0;

**# code for Type B goal 3**

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_minus;

**subject** **to** G1: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s1\_plus + s1\_minus = 5692.9;

**subject** **to** G2: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s2\_plus + s2\_minus = 2 \* 5000;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_plus = 0;

**subject** **to** g2: s2\_minus = 0;

## 

## Appendix G5: AMPL output for Type B GP Model

#goal 1

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

1 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 17643

X3B = 22357

X4B = 10000

X5B = 0

X6B = 0

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 10000

s3\_plus = 25286

s3\_minus = 0

#goal 2

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal2.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

2 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 14976.3

X3B = 25023.7

X4B = 10000

X5B = 0

X6B = 0

X7B = 10000

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 39952.7

s3\_minus = 0

#goal 3

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal3.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

3 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 5000

X3B = 24357

X4B = 10000

X5B = 10643

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

## 

## Appendix G6: SA for Type B GP Model

Goal 1: (Number of 51 > Waste > Number of 24.5)

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s1\_minus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s2\_plus + s2\_minus = 5692.9;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 8869.17

X2B = 5000

X3B = 0

X4B = 10000

X5B = 0

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

Goal 2:

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s2\_plus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s2\_plus + s2\_minus = 5692.9;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal2.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

2 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 14976.3

X3B = 25023.7

X4B = 10000

X5B = 0

X6B = 0

X7B = 10000

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 39952.7

s3\_minus = 0

Goal 3:

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_minus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s2\_plus + s2\_minus = 5692.9;

**subject** **to** G3: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s3\_plus + s3\_minus = 4 \* 5000;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

**subject** **to** g2: s2\_plus = 0;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal3.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

3 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 5000

X3B = 24357

X4B = 10000

X5B = 10643

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

Goal 1: (Number of 51 > Number of 24.5 > Waste)

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s1\_minus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s2\_plus + s2\_minus = 4 \* 5000;

**subject** **to** G3: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s3\_plus + s3\_minus = 5692.9;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 8869.17

X2B = 5000

X3B = 0

X4B = 10000

X5B = 0

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

Goal 2:

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s2\_minus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s2\_plus + s2\_minus = 4 \* 5000;

**subject** **to** G3: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s3\_plus + s3\_minus = 5692.9;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal2.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

0 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 8869.17

X2B = 5000

X3B = 0

X4B = 10000

X5B = 0

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

Goal 3:

**var** X1B >= 0;

**var** X2B >= 0;

**var** X3B >= 0;

**var** X4B >= 0;

**var** X5B >= 0;

**var** X6B >= 0;

**var** X7B >= 0;

**var** X8B >= 0;

**var** s1\_plus >= 0;

**var** s1\_minus >= 0;

**var** s2\_plus >= 0;

**var** s2\_minus >= 0;

**var** s3\_plus >= 0;

**var** s3\_minus >= 0;

**minimize** z: s3\_plus;

**subject** **to** G1: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 0 \*X4B + 0 \*X5B + 1 \*X6B + 1 \*X7B + 1 \*X8B ) - s1\_plus + s1\_minus = 2 \* 5000;

**subject** **to** G2: ( 0 \*X1B + 2 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 2 \*X7B + 2 \*X8B ) - s2\_plus + s2\_minus = 4 \* 5000;

**subject** **to** G3: 0.02 \* (18\*X1B + 15\*X2B + 0\*X3B + 2\*X4B + 15\*X5B + 3\*X6B + 4\*X7B + 14\*X8B) - s3\_plus + s3\_minus = 5692.9;

**subject** **to** C55: ( 0 \*X1B + 0 \*X2B + 3 \*X3B + 3 \*X4B + 0 \*X5B + 2 \*X6B + 2 \*X7B + 1 \*X8B ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1B + 1 \*X2B + 0 \*X3B + 0 \*X4B + 3 \*X5B + 1 \*X6B + 0 \*X7B + 0 \*X8B ) >= 2 \* 5000;

**subject** **to** C49: ( 4 \*X1B + 1 \*X2B + 1 \*X3B + 0 \*X4B + 1 \*X5B + 0 \*X6B + 0 \*X7B + 0 \*X8B ) >= 8 \* 5000;

**subject** **to** C22: ( 0 \*X1B + 0 \*X2B + 0 \*X3B + 1 \*X4B + 0 \*X5B + 0 \*X6B + 0 \*X7B + 2 \*X8B ) >= 2 \* 5000;

**subject** **to** g1: s1\_minus = 0;

**subject** **to** g2: s2\_minus = 0;

ampl: reset;

ampl: option solver cplex;

ampl: model Final\_project\_GP\_B\_goal3.mod;

ampl: solve;

CPLEX 20.1.0.0: optimal solution; objective 0

3 dual simplex iterations (0 in phase I)

ampl: display X1B,X2B,X3B,X4B,X5B,X6B,X7B,X8B,s1\_plus,s1\_minus,s2\_plus,s2\_minus,s3\_plus,s3\_minus;

X1B = 0

X2B = 5000

X3B = 24357

X4B = 10000

X5B = 10643

X6B = 10000

X7B = 0

X8B = 0

s1\_plus = 0

s1\_minus = 0

s2\_plus = 0

s2\_minus = 0

s3\_plus = 0

s3\_minus = 0

## Appendix H: Details of Dynamic Programming Model

In an attempt to optimize the model by minimizing the total cost of producing the required 5000 metal baskets, the team attempted to develop a model and a skill that was outside of the scope of the LP, ILP, and GP models. This model would, in theory, help to arrive at the optimal cost by optimizing the possible cutting patterns simultaneously.

The dynamic programming model used here is one that comes from the famous Knapsack model in the operations research field which involves optimizing the total value of items in a “knapsack” while fulfilling a certain weight requirement. This model, for our purposes, is a secondary model to the primary model of minimizing the total cost while meeting our demand.

The way it works is by starting with an initial set of parameters or cutting patterns and solving the original problem. By considering the cutting pattern parameters as their own variables in the form of a vector, the dual model is solved to maximize the inverse reduced cost of the dual parent model. As such the two models are looped until there are no additional patterns able to enter into the model at which point you would theoretically arrive at a unique optimal solution.

While we were able to provide only 8 possible cutting patterns, we used the output of the dynamic programming model to learn what the optimal patterns might be or simply to confirm if the patterns we were using were optimal. By reviewing the results in Appendix I, it is clear that for Rod type B, the cutting patterns were considered optimal by the reduced cost of the dual model and as such the results of the dynamic model turned out to be identical to the results of the LP mode. However, for Rod Type B, the cutting patterns were not optimal as several other patterns were introduced. Using these newly introduced patterns as a means to learn what might be a better pattern, we modified the patterns of our original ILP model and re-solved the problem to find the model that had the best performance of all the ones analyzed in this report.

Please find the full AMPL formulation and output in Appendix I.

## Appendix I: AMPL for Dynamic Programming Model

### Appendix I.1 AMPL for Dynamic Programming Model Input

#### Data File for Rod A

**param** RodWidth := 184; # width of raw material rod

**param** RodCost := 9.5; # unit cost of raw material rod

**param** WasteCost := 0.02; # cost per inch of waste

# define the requirements for quantity of each piece

**param**: PARTS: Demand Lengths :=

51 10000 51

24.5 20000 24.5

55 20000 55

50 10000 50

49 40000 49

22.5 10000 22.5;

**param** CuttingPatterns :

1 2 3 4 5 6 7 8 :=

51 0 0 0 0 0 1 1 2

24.5 0 3 0 1 2 1 0 1

55 0 2 2 2 2 0 2 1

50 0 0 1 0 0 1 0 0

49 2 0 0 0 0 1 0 0

22.5 3 0 1 2 1 0 1 0

#### Data File for Rod B

**param** RodWidth := 214; # width of raw material rod

**param** RodCost := 11; # unit cost of raw material rod

**param** WasteCost := 0.02; # cost per inch of waste

# define the requirements for quantity of each piece

**param**: PARTS: Demand Lengths :=

51 10000 51

24.5 20000 24.5

55 20000 55

50 10000 50

49 40000 49

22.5 10000 22.5;

**param** CuttingPatterns :

1 2 3 4 5 6 7 8 :=

51 0 0 0 0 0 1 1 1

24.5 0 2 0 1 0 0 2 2

55 0 0 3 3 0 2 2 1

50 0 1 0 0 3 1 0 0

49 4 1 1 0 1 0 0 0

22.5 0 0 0 1 0 0 0 2

#### Model File for Rods A & B

#####

## MASTER PROBLEM

#####

**param** RodWidth > 0; # width of raw material rod (see .dat file)

**param** RodCost > 0; # unit cost of raw material rod (see .dat file)

**param** WasteCost > 0; # cost per inch of waste (see .dat file)

**set** PARTS; # set of PARTS to be cut (ie. 50 in, 51 in, 55 in, etc.)

**param** Demand {PARTS}; # number of each width to be cut (quanty of each part required to meet demand)

**param** Lengths {PARTS};

**param** NumberOfPatterns **integer** >= 0; # number of cutting patterns

**set** PATTERNS := 1..NumberOfPatterns; # set of of all cutting patterns

# parameter for grid of cutting patterns of part i with pattern j

**param** CuttingPatterns {PARTS,PATTERNS} **integer** >= 0;

# check to determine whether the cutting pattern meets the critera to be used (ie. cant cut more than the rod)

**check** {j **in** PATTERNS}:

**sum** {i **in** PARTS} i \* CuttingPatterns[i,j] <= RodWidth;

# waste in inches generated by cutting pattern j

**param** PatternWaste {j **in** PATTERNS} = (RodWidth - **sum** {i **in** PARTS} i \* (CuttingPatterns[i,j]));

# cost of waste in dollars generated by cutting pattern j

**param** PatternWasteCost {j **in** PATTERNS} = WasteCost\*(RodWidth - **sum** {i **in** PARTS} i \* (CuttingPatterns[i,j]));

**var** Cut {PATTERNS} **integer** >= 0; # number of rods cut using pattern j

#param UnusedCost {i in PARTS} = WasteCost \* (((sum {j in PATTERNS} CuttingPatterns[i,j] \* Cut[j]) - Demand[i])\* Lengths[i]);

## OBJECTIVE FUNCTION

### Minimize the total cost

**minimize** Cost {i **in** PARTS}:

RodCost \* (**sum** {j **in** PATTERNS} Cut[j])

+ (**sum** {j **in** PATTERNS} Cut[j] \* PatternWasteCost[j])

+ WasteCost \* (((**sum** {j **in** PATTERNS} CuttingPatterns[i,j] \* Cut[j]) - Demand[i])\* Lengths[i]);

# constraint to meet demand (ie. cut of pattern j including part i >= qty of part i \* 5000)

**subject** **to** DemandConstraint {i **in** PARTS}:

**sum** {j **in** PATTERNS} CuttingPatterns[i,j] \* Cut[j] >= Demand[i];

#####

## SUB PROBLEM - KnapSack

#####

####

**param** price {PARTS} **default** 0.0;

**var** Use {PARTS} **integer** >= 0;

## OBJECTIVE FUNCTION

**maximize** ReducedCost:

**sum** {i **in** PARTS} price[i] \* Use[i] - 1 ;

# Parts i cut from pattern cannot exceed the length of the stock rod

**subject** **to** WidthLimit:

**sum** {i **in** PARTS} Lengths[i] \* Use[i] <= RodWidth;

# Limitation 1: Each metal rails can be cut at most 5 times (into 6 pieces, whether being used or waste)

**subject** **to** MaxNumberOfCuts {j **in** PATTERNS}:

**sum**{i **in** PARTS} CuttingPatterns[i,j] <= 5;

# Limitation 2: At most 8 different patterns can be planned for cutting

**subject** **to** MaxCuttingPatterns:

**max**{j **in** PATTERNS} j <= 8;

#### 

#### Run Script for Rods A & B

#####

## RUN SCRIPT FOR LP

#####

**reset**;

**option** solver cplex;

**option** solution\_round 6;

**model** cut1.mod; # call the model file

**data** cutA.dat; # call the data file (rod A or B)

# define the master problem

**problem** Cutting: Cut, Cost, DemandConstraint;

**option** relax\_integrality 1;

**option** presolve 0;

# define the sub problem

**problem** CutPatternGeneration: Use, ReducedCost, WidthLimit, MaxNumberOfCuts, MaxCuttingPatterns;

**option** relax\_integrality 0;

**option** presolve 1;

**let** NumberOfPatterns := 8;

**repeat** {

**solve** Cutting;

**let** {i **in** PARTS} price[i] := DemandConstraint[i].dual;

**solve** CutPatternGeneration;

**if** ReducedCost > 0.00001 **then** {

**let** NumberOfPatterns := NumberOfPatterns + 1;

**let** {i **in** PARTS} CuttingPatterns[i,NumberOfPatterns] := Use[i];

}

**else** **break**;

};

**display** CuttingPatterns; # show the final cutting patterns used in optimal solution

**display** Cost; # display the objective function value Z

**display** {j **in** PATTERNS} (

Cut[j], # show optimal solution results (cut)

PatternWaste[j], # show pattern waste for optimal solution patterns

PatternWasteCost[j], # show cost of pattern waste for optimal solution values

PatternWasteCost[j]\*Cut[j]); # show the each decision variables contribution to objective function value

**display** {i **in** PARTS} ( WasteCost \* (((**sum** {j **in** PATTERNS} CuttingPatterns[i,j] \* Cut[j]) - Demand[i])\* Lengths[i]) );

### 

### Appendix I.2 AMPL for Dynamic Programming Model Output

#### Output for Dynamic Programming Model Rod Type A

CPLEX 20.1.0.0: optimal solution; objective 326521.4286

6 dual simplex iterations (0 in phase I)

Objective = Cost[51]

CPLEX 20.1.0.0: optimal integer solution; objective 14.481429

2 MIP simplex iterations

0 branch-and-bound nodes

CPLEX 20.1.0.0: optimal solution; objective 262042.8571

1 simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxCuttingPatterns: no variables, but lower bound = -Infinity, upper = -1

presolve, constraint MaxCuttingPatterns: all variables eliminated, but upper bound = -1 < 0

CPLEX 20.1.0.0: optimal solution; objective 262042.8571

0 simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxCuttingPatterns: no variables, but lower bound = -Infinity, upper = -2

presolve, constraint MaxCuttingPatterns: all variables eliminated, but upper bound = -2 < 0

CPLEX 20.1.0.0: optimal solution; objective 262042.8571

0 simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxCuttingPatterns:no variables, but lower bound = -Infinity, upper = -3

presolve, constraint MaxCuttingPatterns:all variables eliminated, but upper bound = -3 < 0

CPLEX 20.1.0.0: optimal solution; objective 262042.8571

0 simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxCuttingPatterns: no variables, but lower bound = -Infinity, upper = -4

presolve, constraint MaxCuttingPatterns:all variables eliminated, but upper bound = -4 < 0

CPLEX 20.1.0.0: optimal solution; objective 262042.8571

0 simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxCuttingPatterns: no variables, but lower bound = -Infinity, upper = -5

presolve, constraint MaxCuttingPatterns:all variables eliminated, but upper bound = -5 < 0

Bailing out after 10 warnings.

CuttingPatterns [\*,\*] (tr)

: 22.5 24.5 49 50 51 55 :=

1 3 0 2 0 0 0

2 0 3 0 0 0 2

3 1 0 0 1 0 2

4 2 1 0 0 0 2

5 1 2 0 0 0 2

6 0 1 1 1 1 0

7 1 0 0 0 1 2

8 0 1 0 0 2 1

9 0 1 3 0 0 0

10 0 1 3 0 0 0

11 0 1 3 0 0 0

12 0 1 3 0 0 0

13 0 1 3 0 0 0

;

Cost [\*] := 262043

;

# $4 = PatternWasteCost[j]\*Cut[j]

: Cut[j] PatternWaste[j] PatternWasteCost[j] $4 :=

1 357.143 18.5 0.37 132.143

2 1071.43 0.5 0.01 10.7143

3 4464.29 1.5 0.03 133.929

4 0 4.5 0.09 0

5 0 2.5 0.05 0

6 5535.71 9.5 0.19 1051.79

7 4464.29 0.5 0.01 44.6429

8 0 2.5 0.05 0

9 11250 12.5 0.25 2812.5

10 0 12.5 0.25 0

11 0 12.5 0.25 0

12 0 12.5 0.25 0

13 0 12.5 0.25 0

;

WasteCost\*((sum{j in PATTERNS} CuttingPatterns[i,j]\*Cut[j] - Demand[i])\*

Lengths[i]) [\*]

:=

22.5 -4.5e-07

24.5 -4.9e-07

49 0

50 0

51 0

55 -2.2e-06

;

#### Output for Dynamic Programming Model Rod Type

CPLEX 20.1.0.0: optimal solution; objective 267200

7 dual simplex iterations (0 in phase I)

Objective = Cost[51]

presolve, constraint MaxNumberOfCuts[8]:

no variables, but lower bound = -Infinity, upper = -1

presolve, constraint MaxNumberOfCuts[8]:

all variables eliminated, but upper bound = -1 < 0

CuttingPatterns [\*,\*] (tr)

: 22.5 24.5 49 50 51 55 :=

1 0 0 4 0 0 0

2 0 2 1 1 0 0

3 0 0 1 0 0 3

4 1 1 0 0 0 3

5 0 0 1 3 0 0

6 0 0 0 1 1 2

7 0 2 0 0 1 2

8 2 2 0 0 1 1

;

Cost [\*] := 267200

;

: Cut[j] PatternWaste[j] PatternWasteCost[j] PatternWasteCost[j]\*Cut[j]

:=

1 9226.19 18 0.36 3321.43

2 0 66 1.32 0

3 0 0 0 0

4 1428.57 2 0.04 57.1429

5 3095.24 15 0.3 928.571

6 714.286 3 0.06 42.8571

7 5000 4 0.08 400

8 4285.71 14 0.28 1200

;

WasteCost\*((sum{j in PATTERNS} CuttingPatterns[i,j]\*Cut[j] - Demand[i])\*

Lengths[i]) [\*]

:=

22.5 4.5e-07

24.5 4.9e-07

49 -9.8e-07

50 -1e-06

51 0

55 1.1e-06

;

#### 

### Appendix I.3 Supplementary ILP Model from DP Model

#### AMPL Input

# INTEGER LINEAR PROGRAMMING MODEL (ILP)

## Given:

# ROD TYPE: A

# ROD LENGTH: 184 inches

# ROD COST: $9.50 per rod

# COST OF WASTE/UNUSED: $0.02 per inch

# define decision variables XiA, i = 1,...,8 is cutting pattern i cut using rod A

**var** X1A **integer** >=0;

**var** X2A **integer** >=0;

**var** X3A **integer** >=0;

**var** X4A **integer** >=0;

**var** X5A **integer** >=0;

**var** X6A **integer** >=0;

**var** X7A **integer** >=0;

**var** X8A **integer** >=0;

**minimize** z: 9.5 \* (X1A + X2A + X3A + X4A + X5A + X6A + X7A + X8A) # cost of type A rods used $9.5 per rod

+ 0.02 \* (18.5\*X1A + 0.5\*X2A + 1.5\*X3A + 4.5\*X4A + 2.5\*X5A + 9.5\*X6A + 0.5\*X7A + 12.5\*X8A) # cost of wasted material from each cut

+ 0.02 \* (

((0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 0 \*X8A) - 2 \* 5000)\* 51.0 + # cost of unused pieces of 51 inches

((0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A) - 4 \* 5000)\* 24.5 + # cost of unused pieces of 24.5 inches

((0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 0 \*X8A) - 4 \* 5000)\* 55.0 + # cost of unused pieces of 55 inches

((0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A) - 2 \* 5000)\* 50.0 + # cost of unused pieces of 50 inches

((2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 3 \*X8A) - 8 \* 5000)\* 49.0 + # cost of unused pieces of 49 inches

((3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A) - 2 \* 5000)\* 22.5 ); # cost of unused pieces of 22.5 inches

# constraints for demand of X inch parts

**subject** **to** C51: ( 0 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C24: ( 0 \*X1A + 3 \*X2A + 0 \*X3A + 1 \*X4A + 2 \*X5A + 1 \*X6A + 0 \*X7A + 1 \*X8A ) >= 4 \* 5000;

**subject** **to** C55: ( 0 \*X1A + 2 \*X2A + 2 \*X3A + 2 \*X4A + 2 \*X5A + 0 \*X6A + 2 \*X7A + 0 \*X8A ) >= 4 \* 5000;

**subject** **to** C50: ( 0 \*X1A + 0 \*X2A + 1 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 0 \*X8A ) >= 2 \* 5000;

**subject** **to** C49: ( 2 \*X1A + 0 \*X2A + 0 \*X3A + 0 \*X4A + 0 \*X5A + 1 \*X6A + 0 \*X7A + 3 \*X8A ) >= 8 \* 5000;

**subject** **to** C22: ( 3 \*X1A + 0 \*X2A + 1 \*X3A + 2 \*X4A + 1 \*X5A + 0 \*X6A + 1 \*X7A + 0 \*X8A ) >= 2 \* 5000;

#### AMPL Output

ampl: reset;

option solver cplex;

option cplex\_options sensitivity; # indicate sensitivity module

model finalreport\_ILP\_RodA.mod;

solve;

display X1A, X2A, X3A, X4A, X5A, X6A, X7A, X8A;

CPLEX 20.1.0.0: sensitivity

CPLEX 20.1.0.0: optimal integer solution within mipgap or absmipgap; objective 262057.92

6 MIP simplex iterations

0 branch-and-bound nodes

absmipgap = 15.0629, relmipgap = 5.74791e-05

suffix up OUT;

suffix down OUT;

suffix current OUT;

X1A = 357

X2A = 1072

X3A = 4464

X4A = 0

X5A = 0

X6A = 5536

X7A = 4465

X8A = 11250

## Appendix J: List of Variables

| **No.** | **Model** | **Variable** | **Description** |
| --- | --- | --- | --- |
| 1 | LP, ILP | XiA | Cut pattern i, i = 1,...,8 for rod type A   * LP: All variables ≥ 0 * ILP: All variables integer ≥ 0 |
| 2 | LP, ILP | XiB | Cut pattern i, i = 1,...,8 for rod type B   * LP: All variables ≥ 0 * ILP: All variables integer ≥ 0 |
| 3 | GP | SiP | Positive deviation i=1, 2, 3 |
| 4 | GP | SiN | Negative deviation i=1, 2, 3 |
| 5 | GP | S1P | Amount of the dollars that exceeds targeted waste cost from extra pieces wasted in goal 1. This is the undesirable variable for G1. |
| 6 | GP | S1N | Amount of the dollars that is below targeted waste cost from extra pieces wasted in goal 1 |
| 7 | GP | S2P | Number of 51 inch pieces that exceeds goal 2. |
| 8 | GP | S2N | Number of 51 inch pieces that below goal 2. This is the undesirable variable for G2. |
| 9 | GP | S3P | Number of 24.5 inch pieces that exceeds goal 3. |
| 10 | GP | S3N | Number of 24.5 inch pieces that exceeds goal 3. This is the undesirable variable for G3. |
| 11 | DP | RodWidth | Overall width of Rod (A: 184 in. B: 214 in.) |
| 12 | DP | RodCost | Cost per Rod (A: $9.50 B: $11.00) |
| 13 | DP | WasteCost | Cost of waste $/inch = $0.02/inch |
| 14 | DP | PARTS | A set of the number parts required |
| 15 | DP | Demand[i] | The amount of each part required |
| 16 | DP | Lengths[i] | A parameter of the lengths (same as PARTS) |
| 17 | DP | NumberOfPatterns | Integer value of total number of cutting patterns |
| 18 | DP | CuttingPatterns[i,j] | Matrix of cutting patterns of part i and pattern j |
| 19 | DP | Cut[j] | Qty. of rods to be cut using pattern j |
| 20 | DP | PatternWaste[j] | Waste in inches for each pattern j |
| 21 | DP | PatternWasteCost[j] | Waste in dollars for each pattern j |

## Appendix K: Problems & Suggestions for Future

### Appendix K.1 Problems

* We use Python to list all the possible patterns, use 8 patterns with least waste for type A and choose 8 patterns with no waste rob for type B. We didn’t try other patterns and maybe there exists a better combination.
* Choosing the rod with 51 and 24.5 inches as the most important two, is kind of unconvincing.
* The GP result for type B rod is significantly higher than the LP solution in terms of total cost and number of rods to be purchased. Possibly because of the lack of goal or a constraint on the total cost. Therefore, the cplex software increased the number of rods and hence we cut a lot of extra 55 inches to meet the goals.

### Appendix K.2 Suggestions

* Try more patterns with different combinations to see if there is another way to make the total cost smaller.
* There is a terminology called margin cost, which means the extra cost you need to produce one unit of new product. We plan to explore if the results would change for producing more brackets. And then try to find the range and the combination with the smallest margin cost. If we have more information like the tax rate, the price and the shipment fee of the bracket, we will try to find how to maximize the company’s profit.
* Explore whether we could further minimize the total cost with more different patterns like 9 or 10.
* We should have goals of minimizing total cost as well or constraints on the number of rods that can be purchased, to avoid purchasing extra rods to meet the goals.

## Appendix L: Project Evaluation

| **Project Evaluation Sheet**  **Date:** 8 December 2021  **Course:** ISE 536  **Project Name:** Final Project Report   | **Team Member Name** | **% of the project** | **Explanation** | **Signature** | | --- | --- | --- | --- | | Daniel Ley | 28% | LP model intro, LP sensitivity analysis, ILP model (all parts, code, and appendices), Dynamic Model (all parts, code, and appendices), supplementary ILP model from dynamic Model, Report to Manager (all parts), Project overview, Appendix J - list of variables |  | | Ruoqian Lan | 25% | GP model problem statement and goals. Summary table of the GP type A. Appendix F: Details of GP formulation and explanations. Summary of results from the GP type A output and analysis of the solution. GP sensitivity Analysis for the GP type A. Summary table of solutions. Compare output of LP and GP for type A, provide comments. Summary table and input information for LP. |  | | Vaishnavi Guntupalli | 23% | LP introduction, Summary table fo input information for GP type B, Summary results of type B output, Analysis of solution, GP sensitivity analysis for GP type B, Comparison of LP and GP outputs for type B, Appendix F: formulation of GP for type B Appendix G4, G5,G6,K |  | | Shuochen Tao | 24% | change the LP word part to fit the new optimal solution, LP sensitivity analysis(code part only), GP code and sensitivity analysis code(Type A only), Appendix A.2 and K |  |   **Comments:** Good cooperation. Everyone is willing to help and kind. |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |