IE 535 Project

Saichander Reddy Naini
PUID: 00322317861

December 2021

Model 5 and Model 22 (extra credit) are solved.

Model 5

x_{ij}: Weight of food i for type j.
$i \in \{1, 2, 3, 4\}$
$j \in \{1, 2\}$

Objective

$$\min -0.895x_{11} - 1.305x_{21} - 1.265x_{31} - 1.125x_{41} - 0.37x_{12} - 0.78x_{22} - 0.74x_{32} - 0.6x_{42}$$

s.t

$$43.419x_{11} + 6.56x_{21} + 11.41x_{31} + 25.96x_{41} \geq 0$$
$$11.075x_{11} + 35.785x_{21} + 23.435x_{31} + 37.685x_{41} \leq 0$$
$$3.579x_{11} + 0.879x_{21} + 1.779x_{31} + 2.679x_{41} \geq 0$$
$$11.075x_{12} + 35.785x_{22} + 23.435x_{32} + 37.685x_{42} \geq 0$$
$$x_{11} + x_{12} \leq 1500$$
$$x_{21} + x_{22} \leq 500$$
$$x_{31} + x_{32} \leq 1000$$
$$x_{41} + x_{42} \leq 2000$$
$$x_{ij} \geq 0 \ \forall i, j$$

Code

```python
import numpy as np



def Phase_1(A, b, z):
    m, n = A.shape
    I = np.identity(m)
    # Adds artificial variables to all the constraints i.e "checks" for the identity part
    A_new = np.hstack((A, I))
    cb = np.array([1 for _ in range(m)])
    z_new = cb@I@A          # C_N is zero in 2 phase objective
```

```
        padding = np.array([0 for _ in range(m)])
        z_new = np.hstack((z_new, padding))
        xb1 = np.array([(n+i+1) for i in range(m)])
        z_o = cb@I@b
        # building the tableau for 2 phase
        tb, m_2, n_2 = build_tableau(A_new, b, -z_new, xb1, z_o)
        # Phase 1 of Simplex
        tb, z_star, sol, xb = Simplex(tb, m_2, n_2, opt_flag= False)
        if z_star != 0:
            print("Infeasible")
            return (0, 0, 0, 0)
        # Removing the artificial variables
        artf_idx = list(range(n+m+2))
        del artf_idx[-(m+1):-1]
        tb = tb[:, artf_idx]
        # Redundent Constraints
        l = [np.where(xb==i)[0] for i in xb1]
        for i in l:
            tb = np.delete(tb, i+2, axis = 0)
            xb = np.delete(xb, i, axis=0)
            sol = np.delete(sol, i, axis = 0)
        m = xb.shape[0]
        # Convert the  2 phase z row to regular z row
        tb[0, :] = np.array([i for i in range(n+2)])
        nbv_idx = np.array(list(range(1, n+1)))
        nbv_idx = np.setdiff1d(nbv_idx, xb)
        B_invN = tb[2:, nbv_idx]
        cb = z[(xb-1).astype(int)]
        cn = z[(nbv_idx-1).astype(int)]
        tb[1, nbv_idx.astype(int)] = cb@B_invN - cn
        tb[1, xb.astype(int)] = 0
        tb[1, -1] = cb@sol
        # Phase II Simplex
        f_tb, f_z_star, f_sol, f_xb = Simplex(tb, m, n)
        return f_tb, f_z_star, f_sol, f_xb



    def Optimality_check(z, xb):
        #Bland's rule: argmax picks the smallest index for a tie
        idx = np.argmax(z)
        if z[idx] == 0:
            return True, idx+1
        return False, idx+1

    def ratio_test(b, pivot_col, xb):
        ix = 0
        flag2 = False
        if pivot_col[np.argmax(pivot_col)] <= 0:
            flag2 = True
            return flag2, ix+2
        pivot_col[pivot_col<0] = 0
        np.seterr(divide='ignore')
        t = b/pivot_col
        t[np.isnan(t)] = np.Inf
        # t[t<0] = np.Inf
```

```
        ix = np.argmin(t)
        #Bland's rule: picks the smallest index for a tie
        bld_rule = (t==t[ix])
        if sum(bld_rule) > 1:
            ties = xb[bld_rule]
            min_idx = np.argmin(ties)
            ix = np.where(xb==ties[min_idx])[0][0]
        return flag2, ix+2



    def build_tableau(A, b, z, xb, z_o = 0):
        m, n = A.shape
        tableau = np.zeros((m+2, n+2))
        tableau[0,:] = np.array(range(n+2))
        tableau[1, 1:n+1] = -z
        tableau[1, -1] = z_o
        tableau[2:(m+2), 1:n+1] = A
        tableau[2:(m+2), -1] = b
        tableau[2:, 0] = xb
        return tableau, m, n

    def Simplex(tableau, m, n, opt_flag = True):
        i = 0
        while 1:
            eps = 1e-6
            tableau[np.abs(tableau) < eps] = 0
            z = tableau[1, 1:n+1]
            xb = tableau[2:, 0]
            b = tableau[2:(m+2), -1]
            flag1, idx = Optimality_check(z, xb)
            if flag1 == True:
                if opt_flag == True:
                    print("Optimal Solution.")
                break
            pivot_col = np.copy(tableau[2:, idx])
            flag2, ix = ratio_test(b, pivot_col, xb)
            if flag2 == True:
                print("Unbounded.")
                break
            pivot = tableau[ix, idx]
            tableau[ix, 1:] = tableau[ix, 1:]/pivot
            npvt = list(range(1, m+2))
            npvt.remove(ix)
            pivot_col_ex_pivot = tableau[npvt, idx]
            pivot_col_ex_pivot.shape = (m,1)
            pivot_row = tableau[ix, 1:]
            tableau[npvt, 1:] = tableau[npvt, 1:] - (pivot_col_ex_pivot * pivot_row) #broadcasting works
            tableau[ix, 0] = idx
            i = i+1
            print("tableau for iteration {0}: \n".format(i),tableau)


        z_star =  tableau[1, -1]
        sol = b
        xb = tableau[2:, 0]
```

```
124
125         return tableau, z_star, sol, xb
126
127
128
129
130     if __name__ == '__main__':
131
132         # TEST CASES
133
134         # Regular problem for testing
135         # P = [[1, 1, -4, 0, 0, 0],
136         #      [1, 1, 2, 1, 0, 0],
137         #      [1, 1, -1, 0, 1, 0],
138         #      [-1, 1, 1, 0, 0, 1]]
139
140         # b = [9, 2, 4]
141
142         # xb = [4, 5, 6]
143
144         # Two phase problem for testing
145         # P = [[6, 3, 0, 0, 0],
146         #      [1, 1, -1, 0, 0],
147         #      [2, -1, 0, -1, 0],
148         #      [0, 3, 0, 0, 1]]
149
150         # b = [1, 1, 2]
151
152         # xb = [4, 5, 6]
153
154         # P = [[-3, -2, -1, 0, 0],
155         #      [3, -3, 2, 1, 0],
156         #      [-1, 2, 1, 0, 1]]
157
158         # b = [3, 6]
159
160         # xb = [4, 5]
161
162         # Edge case with 0/-ve in ratio test
163         # P = [[-1, -3, 0, 0, 0],
164         #      [1, -2, 1, 0, 0],
165         #      [-2, 1, 0, 1, 0],
166         #      [5, 3, 0, 0, 1]]
167
168         # b = [0, 4, 15]
169
170         # xb = [3, 4, 5]
171
172         # Redundent constraint problem
173
174         # P = [[19, 17, 23, 21, 25],
175         #      [60, 25, 45, 20, 50],
176         #      [10, 15, 45, 50, 40],
177         #      [30, 60, 10, 30, 10],
178         #      [1, 1, 1, 1, 1]]
179
```

```
180         # b = [40, 35, 25, 1]
181
182         # degenerate LP
183
184         # P = [[-1, -1, -1, 0, 0],
185         #      [1, 1, 0, 1, 0],
186         #      [0, -1, 1, 0, 1]]
187
188         # b = [8, 0]
189
190         # PROJECT MODELS
191
192         # Model 5
193         #['x11', 'x21', 'x31', 'x41', 'x12', 'x22', 'x32', 'x42', 's1', 's2', 's3', 's4', 's5', 's6', 's7', 's8']
194
195         #P = [[-0.895, -1.305, -1.265, -1.125, -0.37, -0.78, -0.74, -0.6, 0, 0, 0, 0, 0, 0, #0, 0],
196         #    [43.419, 6.56, 11.41, 25.96, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0],
197         #    [11.075, 35.785, 23.435, 37.685, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0],
198         #    [3.579, 0.879, 1.779, 2.679, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0],
199         #    [0, 0, 0, 0, 11.075, 35.785, 23.435, 37.685, 0, 0, 0, -1, 0, 0, 0, 0],
200         #    [1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0],
201         #    [0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0],
202         #    [0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0],
203         #    [0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1]]
204
205         #b = [0, 0, 0, 0, 1500, 500, 1000, 2000]
206
207         # EXTRA CREDIT: MODEL 22
208         #[x111, x121, x131, x141, x151, x211, x221, x231, x241, x251, x311, x321, x331, x341, x351, x112, x122, x132, x142, x152, x2
209         P = [[61, 72, 45, 55, 66, 69, 78, 60, 49, 56, 59, 66, 63, 61, 47, 58.5, 68.3, 47.8, 0, 63.5, 65.3, 74.8, 55.0, 49.0,
210         57.5, 0, 61.3, 63.5, 58.8, 50.0, 0, 0, 0, 0],
211             [1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0],
212             [0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 0],
213             [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 1, 0],
214             [1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0],
215             [0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0],
216             [0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0],
217             [0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
218             [0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0],
219             [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 27.5, 30.3, 23.8, 0, 28.5, 29.3, 31.8, 27.0, 25.0, 26.5, 0, 28.3,
220             27.5, 26.8, 24.0, 0, 0, 0, 1],
221             [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
222             [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0]]
223
224         b = [10, 20, 15, 7, 11, 9, 10, 8, 675000, 0, 0]
225
226         P = np.array(P)
227         b = np.array(b)
228         z = P[0, :]
229         A = P[1:, :]
230         # xb = np.array(xb)
231
232         print("Input matrix A:{0}\nb:{1}".format(A,  b.T))
233
234         # A, b, z have there usual meaning. Input has to be in standard form (minimization)
235         # with slack variables. For the initial basis program automatically adds artificial
```

```
    # variables for all the constraints and every problem goes through two phases.
    #
    # The code that deals with redundancy is in "Phase_1" function
    #
    # Bland's rule is used to prevent cycling. Check Ratio_test, Optimality_check #functions

    f_tb, f_z_star, f_sol, f_xb = Phase_1(A, b, z)

    print("Final Optimal Tableau:", f_tb)

    print("Optimal Objective function value:", f_z_star)
    for i, j in zip(f_xb, f_sol):
        print("Var:{0} = {1}".format(i, j))
    print("Rest of the variables are non basic variables, therfore = 0")
```

Results

NOTE: Output for each iteration is in the Appendix!

$$\begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \\ x_{41} \\ x_{12} \\ x_{22} \\ x_{32} \\ x_{42} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \\ s_8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1500 \\ 500 \\ 1000 \\ 2000 \\ 0 \\ 0 \\ 0 \\ 133310.0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$z^* = -2884.9999$$

```
Optimal Objective function value: -2884.9999999999995
Var:1.0 = 0.0
Var:9.0 = 0.0
Var:7.0 = 1000.0
Var:5.0 = 1500.0
Var:6.0 = 500.0
Var:12.0 = 133310.0
Var:8.0 = 2000.0
Rest of the variables are non basic variables, therfore = 0
```

Figure 1: Optimal Solution

```
Optimal Solution.
Final Optimal Tableau: [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
   4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
   8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
   1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
   1.60000000e+01  1.70000000e+01]
 [ 0.00000000e+00  0.00000000e+00 -1.17135440e+00 -5.85914221e-01
  -1.26142212e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -4.74040632e-02  0.00000000e+00
   0.00000000e+00 -3.70000000e-01 -7.80000000e-01 -7.40000000e-01
  -6.00000000e-01 -2.88500000e+03]
 [ 1.00000000e+00  1.00000000e+00  3.23115124e+00  2.11602709e+00
   3.40270880e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  9.02934537e-02  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 9.00000000e+00  0.00000000e+00  1.33733356e+02  8.04657801e+01
   1.21782214e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.00000000e+00  3.92045147e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 7.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  1.00000000e+03]
 [ 5.00000000e+00  0.00000000e+00 -3.23115124e+00 -2.11602709e+00
  -3.40270880e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -9.02934537e-02  0.00000000e+00
   0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.50000000e+03]
 [ 6.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  5.00000000e+02]
 [ 1.20000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -1.00000000e+00  0.00000000e+00
   1.00000000e+00  1.10750000e+01  3.57850000e+01  2.34350000e+01
   3.76850000e+01  1.33310000e+05]
 [ 8.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  2.00000000e+03]]
```

Figure 2: Optimal tableau

Commercial Solver (Octave gplk)

```
c = [-0.895, -1.305, -1.265, -1.125, -0.37, -0.78, -0.74, -0.6, 0, 0, 0, 0, 0, 0, 0, 0]';

A = [43.419, 6.56, 11.41, 25.96, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0;
        11.075, 35.785, 23.435, 37.685, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0;
        3.579, 0.879, 1.779, 2.679, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0;
        0, 0, 0, 0, 11.075, 35.785, 23.435, 37.685, 0, 0, 0, -1, 0, 0, 0, 0;
        1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0;
        0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
        0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0;
        0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1];

% A = [ 1, 1, 1;
%      10, 4, 5;
%       2, 2, 6];
b = [0, 0, 0, 0, 1500, 500, 1000, 2000]';
lb = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]';
ub = [];
ctype = "SSSSSSSS";
vartype = "CCCCCCCCCCCCCCCC";
s = 1;

param.msglev = 1;
param.itlim = 100;

[xmin, fmin, status, extra] = ...
   glpk (c, A, b, lb, ub, ctype, vartype, s, param);

disp("Status"), disp(status);
disp("Optimal Objective:"), disp(fmin);
disp("Solution:"), disp(xmin);
```

$$\begin{bmatrix} x_{11} \\ x_{21} \\ x_{31} \\ x_{41} \\ x_{12} \\ x_{22} \\ x_{32} \\ x_{42} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \\ s_8 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1500 \\ 500 \\ 1000 \\ 2000 \\ 0 \\ 0 \\ 0 \\ 133310.0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$z^* = -2885$$

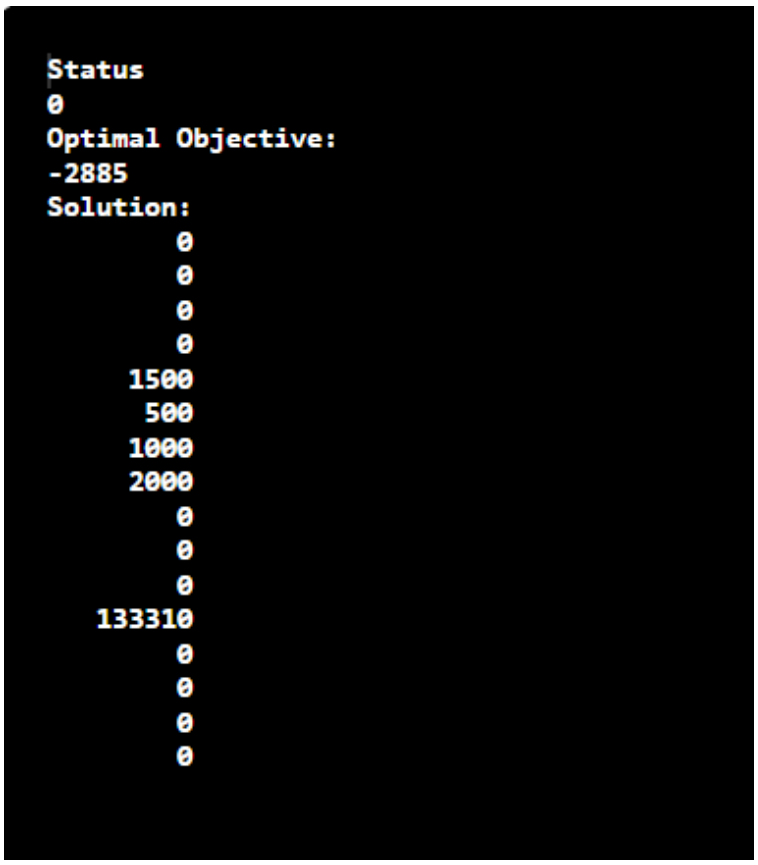

Figure 3: Commercial Solver Output

Extra Credit: Model 22

$i \in \{1, 2, 3\}$
$j \in \{1, 2, 3, 4, 5\}$
$k \in \{1, 2\}$
r_{ij}: Unit cost by rail for ith source and jth market
s_{ij}: Unit cost by ship for ith source and jth market
I_{ij}: Uniform annual cost of investment for ith source and jth market
x_{ijk}: wood units in million board feet for ith source, jth market and kth transportation type
a_i: Availability of ith source
d_j: Demand for jth market

Objective

$$\min \sum_j \sum_i r_{ij} x_{ij1} + \sum_j \sum_i (s_{ij} + I_{ij}) x_{ij2}$$

s.t

$$\sum_k \sum_j x_{ijk} \leq a_i \ \forall i$$

$$\sum_k \sum_i x_{ijk} = d_j \ \forall j$$

$$\sum_j \sum_i I_{ij} x_{ij2} \leq 675000$$

$$x_{142} = 0$$

$$x_{312} = 0$$

$$x_{ijk} \geq 0$$

Results

NOTE: Output for each iteration is in the Appendix at the end!

$$x_{131} = 9.0,\ x_{241} = 10.0,\ x_{251} = 4.0,\ x_{351} = 4.0,\ x_{112} = 1.0,\ x_{212} = 6.0,\ x_{322} = 11,\ s3 = 674485.4$$

$$z^* = 2431.5999$$

```
Optimal Objective function value: 2431.599999999999
Var:21.0 = 6.0
Var:27.0 = 11.0
Var:15.0 = 4.0
Var:16.0 = 1.0
Var:9.0 = 10.0
Var:32.0 = 0.0
Var:10.0 = 4.0
Var:3.0 = 9.0
Var:34.0 = 674485.4000000001
Var:19.0 = 0.0
Var:26.0 = 0.0
Rest of the variables are non basic variables, therfore = 0
```

Figure 4: Model 22 Optimal value

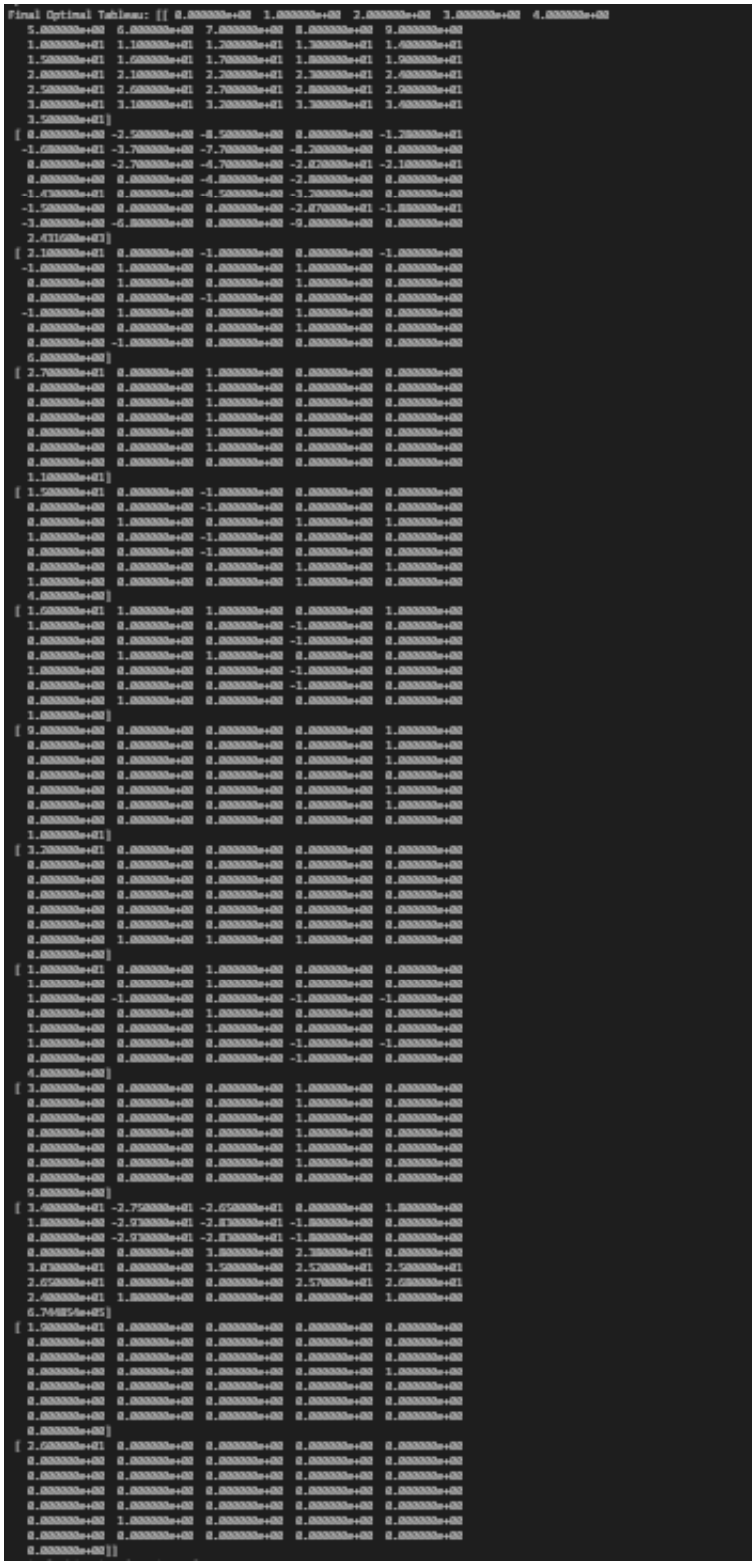

Figure 5: Model 22 Optimal Tableau

Commercial Solver (Octave gplk)

```
c = [-0.895, -1.305, -1.265, -1.125, -0.37, -0.78, -0.74, -0.6, 0, 0, 0, 0, 0, 0, 0, 0]';

A = [43.419, 6.56, 11.41, 25.96, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0;
        11.075, 35.785, 23.435, 37.685, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0;
        3.579, 0.879, 1.779, 2.679, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0;
        0, 0, 0, 0, 11.075, 35.785, 23.435, 37.685, 0, 0, 0, -1, 0, 0, 0, 0;
        1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0, 0;
        0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 0;
        0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0;
        0, 0, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1];

% A = [ 1, 1, 1;
%      10, 4, 5;
%       2, 2, 6];
b = [0, 0, 0, 0, 1500, 500, 1000, 2000]';
lb = [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]';
ub = [];
ctype = "SSSSSSSS";
vartype = "CCCCCCCCCCCCCCCC";
s = 1;

param.msglev = 1;
param.itlim = 100;

[xmin, fmin, status, extra] = ...
   glpk (c, A, b, lb, ub, ctype, vartype, s, param);

disp("Status"), disp(status);
disp("Optimal Objective:"), disp(fmin);
disp("Solution:"), disp(xmin);
```

$$x_{131} = 9.0,\ x_{241} = 10.0,\ x_{251} = 4.0,\ x_{351} = 4.0,\ x_{112} = 1.0,\ x_{212} = 6.0,\ x_{322} = 11,\ s3 = 674485.4$$

$$z^* = 2431.6$$

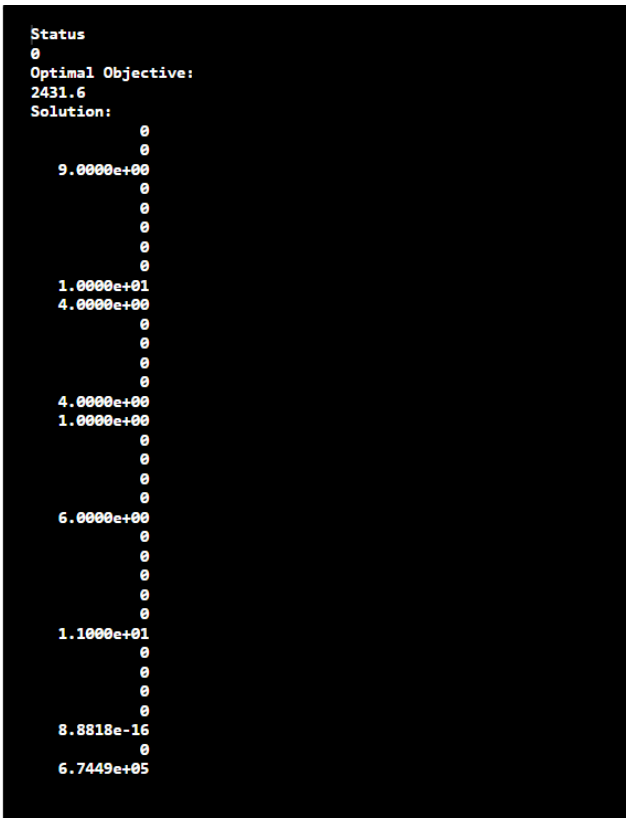

Figure 6: Solution of Commercial Solver

Appendix

Model 5 tableau Output

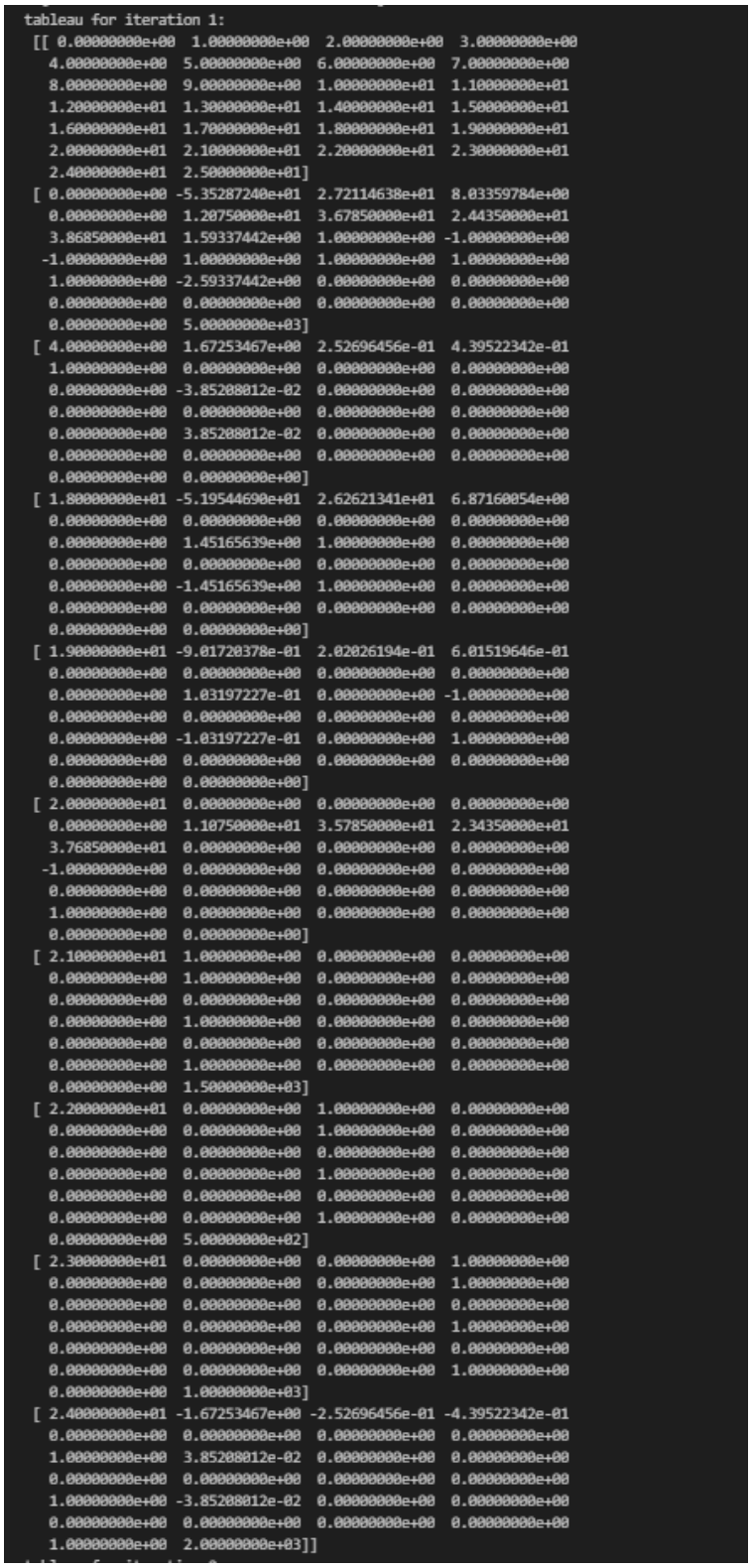

Figure 7: Phase 1 itr 1

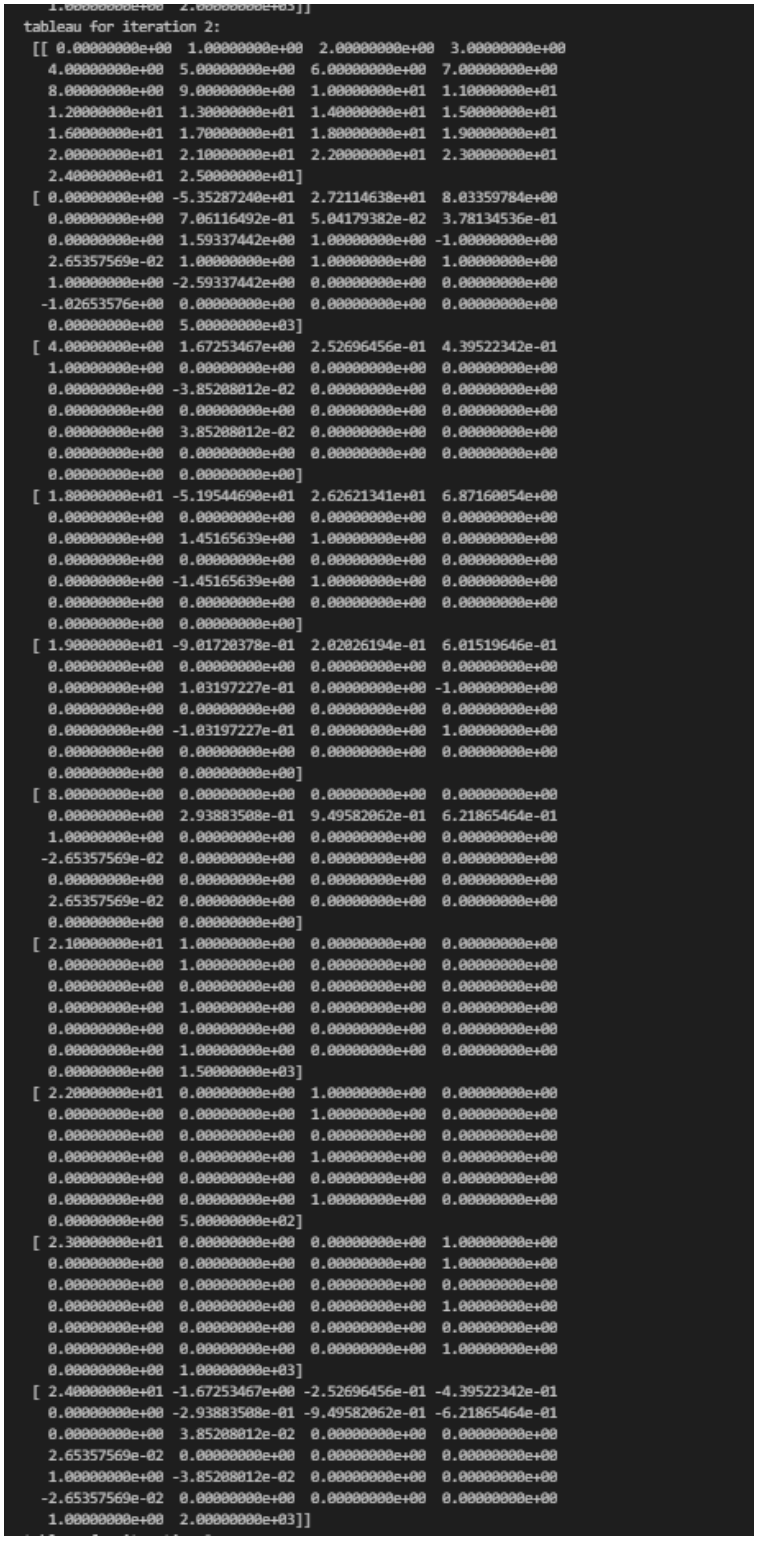

Figure 8: Phase 1 itr 2

```
tableau for iteration 3:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
    4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
    8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
    1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
    1.60000000e+01  1.70000000e+01  1.80000000e+01  1.90000000e+01
    2.00000000e+01  2.10000000e+01  2.20000000e+01  2.30000000e+01
    2.40000000e+01  2.50000000e+01]
  [ 0.00000000e+00 -2.33634600e+02  0.00000000e+00 -3.92960976e+01
   -1.07684390e+02  7.06116492e-01  5.04179382e-02  3.78134536e-01
    0.00000000e+00  5.74146341e+00  1.00000000e+00 -1.00000000e+00
    2.65357569e-02  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00 -6.74146341e+00  0.00000000e+00  0.00000000e+00
   -1.02653576e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  5.00000000e+03]
  [ 2.00000000e+00  6.61875000e+00  1.00000000e+00  1.73932927e+00
    3.95731707e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00 -1.52439024e-01  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.52439024e-01  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 1.80000000e+01 -2.25776969e+02  0.00000000e+00 -3.88068979e+01
   -1.03927591e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  5.45503049e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00 -5.45503049e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 1.90000000e+01 -2.23888125e+00  0.00000000e+00  2.50129573e-01
   -7.99481707e-01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.33993902e-01  0.00000000e+00 -1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00 -1.33993902e-01  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 8.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  2.93883508e-01  9.49582062e-01  6.21865464e-01
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   -2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 2.10000000e+01  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.50000000e+03]
  [ 2.20000000e+01 -6.61875000e+00  0.00000000e+00 -1.73932927e+00
   -3.95731707e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  1.52439024e-01  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00 -1.52439024e-01  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  5.00000000e+02]
  [ 2.30000000e+01  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  1.00000000e+03]
  [ 2.40000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00 -2.93883508e-01 -9.49582062e-01 -6.21865464e-01
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   -2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  2.00000000e+03]]
```

Figure 9: Phase 1 itr 3

```
   1.00000000e+00  2.00000000e+03]]
tableau for iteration 4:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
    4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
    8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
    1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
    1.60000000e+01  1.70000000e+01  1.80000000e+01  1.90000000e+01
    2.00000000e+01  2.10000000e+01  2.20000000e+01  2.30000000e+01
    2.40000000e+01  2.50000000e+01]
  [ 0.00000000e+00  3.99747352e+00  0.00000000e+00  1.54847422e+00
    1.70023474e+00  7.06116492e-01  5.04179382e-02  3.78134536e-01
    0.00000000e+00  0.00000000e+00 -5.25080341e-02 -1.00000000e+00
    2.65357569e-02  1.00000000e+00  1.00000000e+00  1.00000000e+00
    1.00000000e+00 -1.00000000e+00 -1.05250803e+00  0.00000000e+00
   -1.02653576e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  5.00000000e+03]
  [ 2.00000000e+00  3.09487215e-01  1.00000000e+00  6.54883331e-01
    1.05309487e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  2.79446696e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  2.79446696e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 9.00000000e+00 -4.13887639e+01  0.00000000e+00 -7.11396535e+00
   -1.90516976e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  1.83317032e-01  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00 -1.00000000e+00  1.83317032e-01  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 1.90000000e+01  3.30696074e+00  0.00000000e+00  1.20335755e+00
    1.75332961e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -2.45633645e-02 -1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -2.45633645e-02  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 8.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  2.93883508e-01  9.49582062e-01  6.21865464e-01
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   -2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
  [ 2.10000000e+01  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.50000000e+03]
  [ 2.20000000e+01 -3.09487215e-01  0.00000000e+00 -6.54883331e-01
   -1.05309487e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -2.79446696e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -2.79446696e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  5.00000000e+02]
  [ 2.30000000e+01  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  1.00000000e+03]
  [ 2.40000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00 -2.93883508e-01 -9.49582062e-01 -6.21865464e-01
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   -2.65357569e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  2.00000000e+03]]
 tableau for iteration 5:
```

Figure 10: Phase 1 itr 4

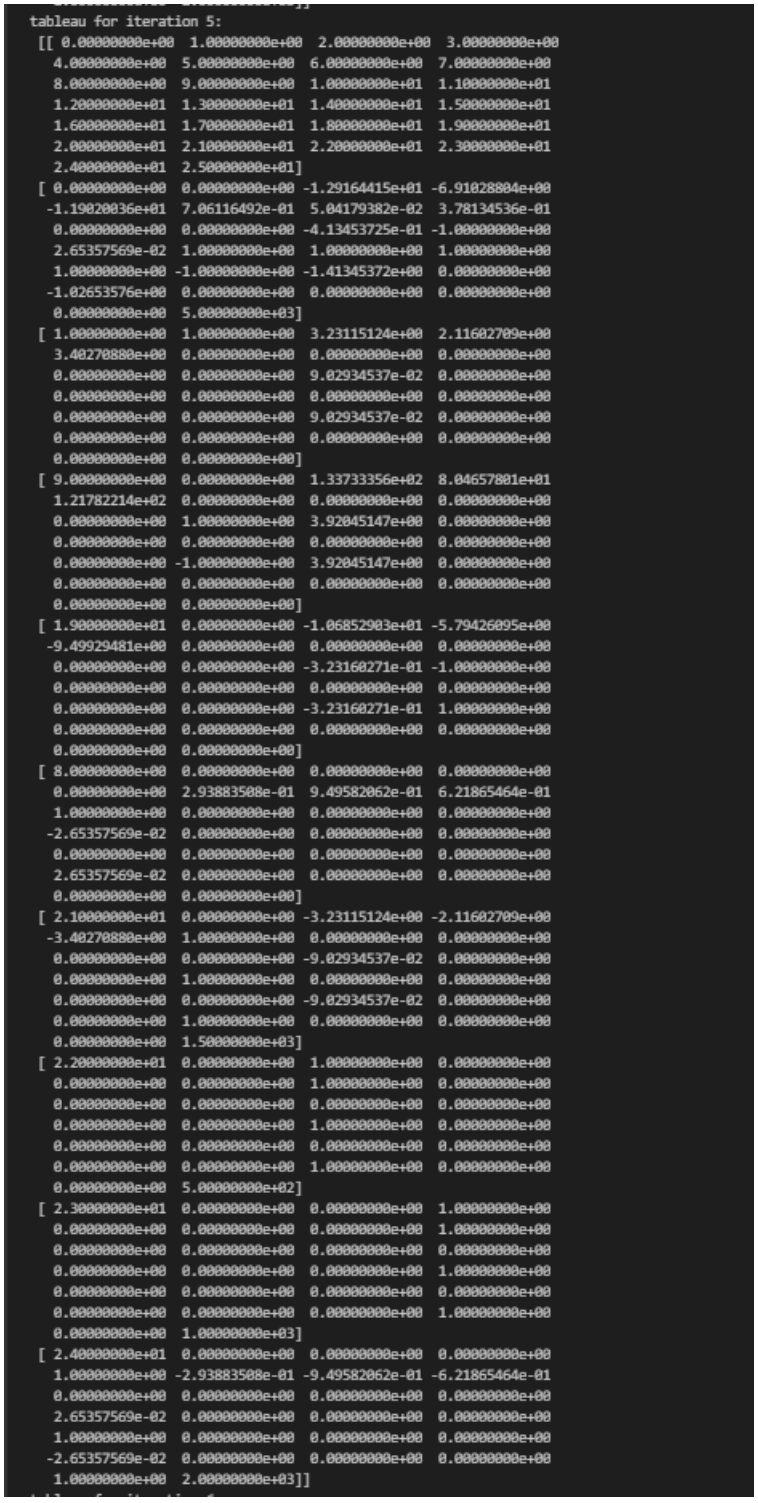

Figure 11: Phase 1 itr 5

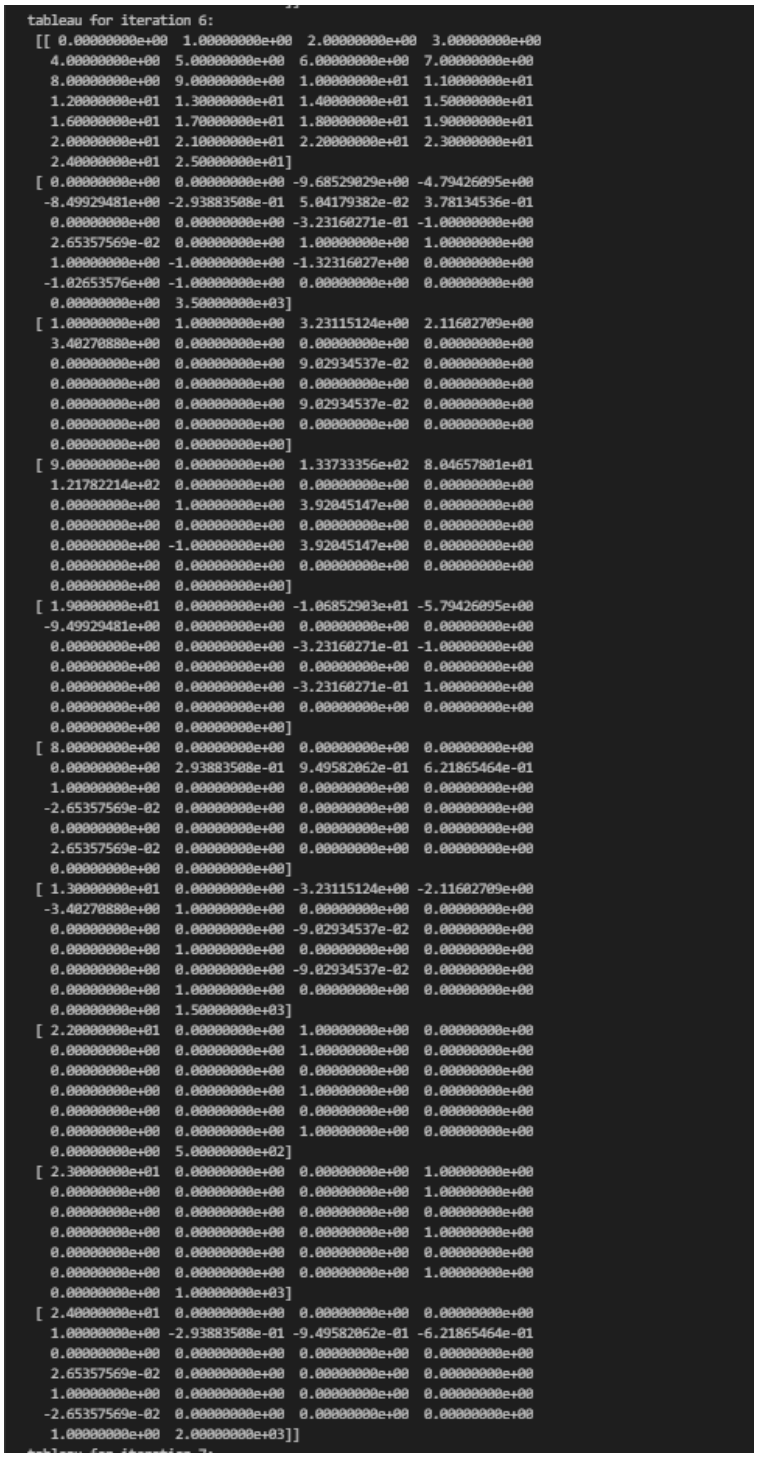

Figure 12: Phase 1 itr 6

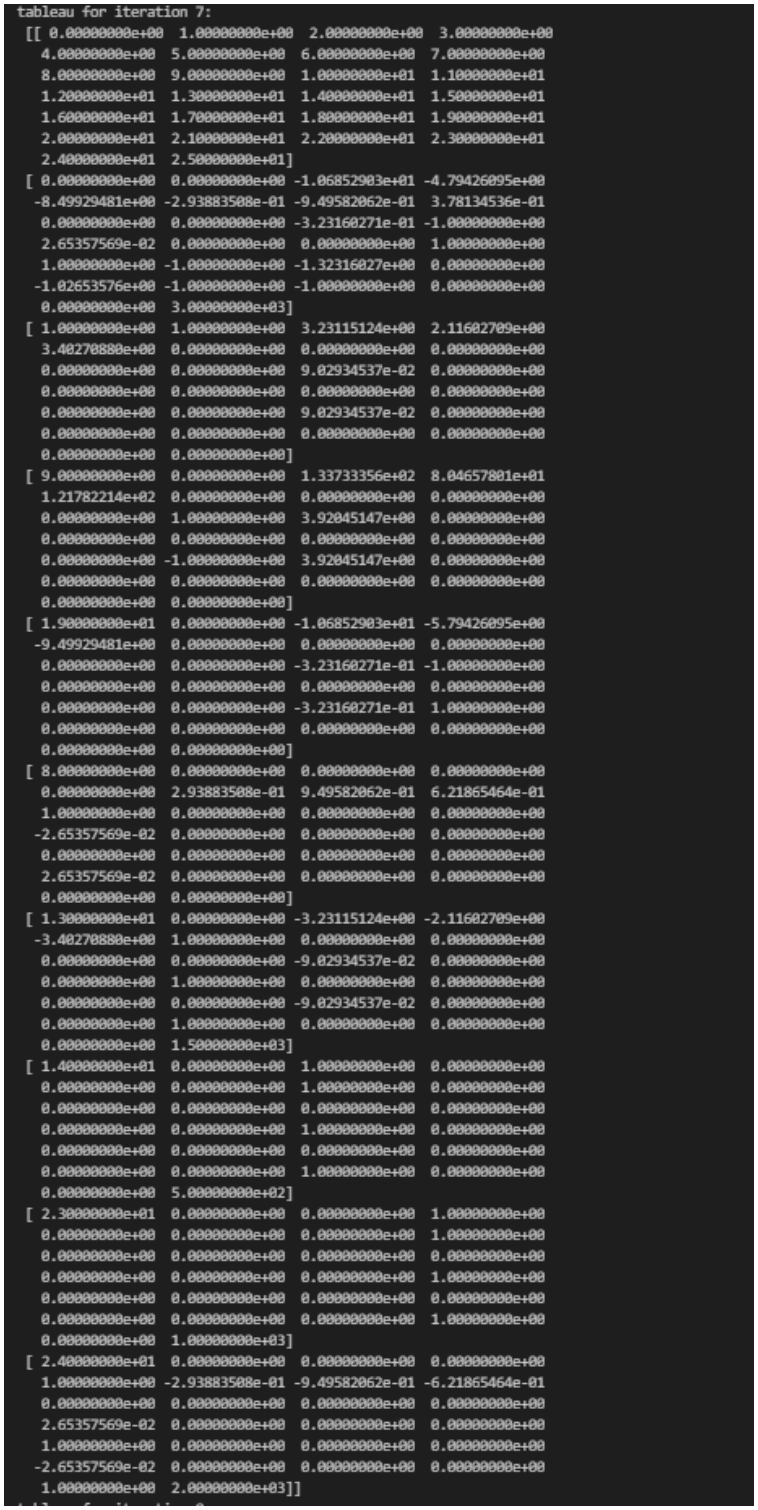

Figure 13: Phase 1 itr 7

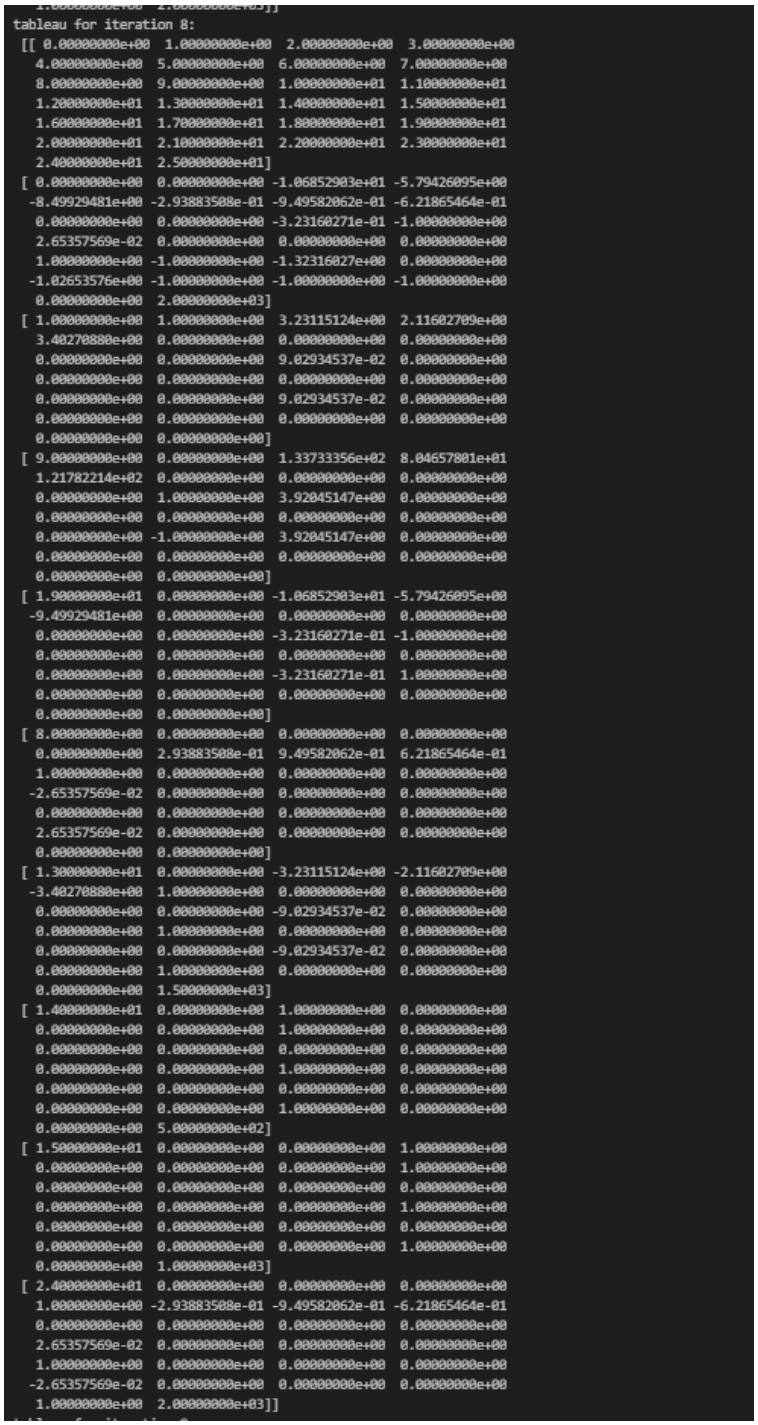

Figure 14: Phase 1 itr 8

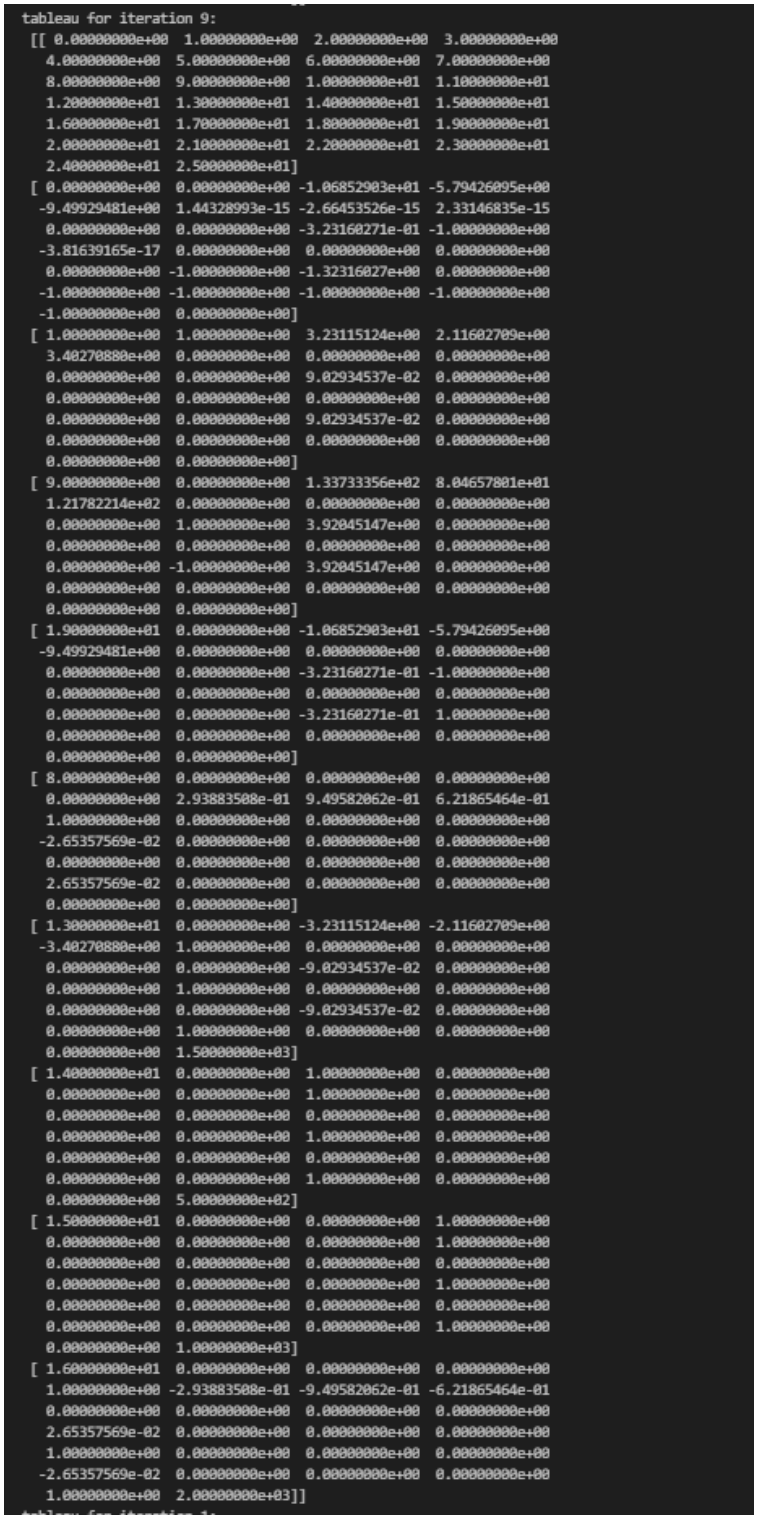

Figure 15: Phase 1 itr 9

```
tableau for iteration 1:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
   4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
   8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
   1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
   1.60000000e+01  1.70000000e+01]
 [ 0.00000000e+00  0.00000000e+00 -1.58688036e+00 -6.28844244e-01
  -1.92042438e+00  2.02880307e-02 -3.49972264e-01  0.00000000e+00
  -5.89967997e-01  0.00000000e+00 -8.08126411e-02  0.00000000e+00
   3.15767015e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 1.00000000e+00  1.00000000e+00  3.23115124e+00  2.11602709e+00
   3.40270880e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  9.02934537e-02  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 9.00000000e+00  0.00000000e+00  1.33733356e+02  8.04657801e+01
   1.21782214e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.00000000e+00  3.92045147e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 7.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  4.72583742e-01  1.52698955e+00  1.00000000e+00
   1.60806486e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
  -4.26712183e-02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 1.30000000e+01  0.00000000e+00 -3.23115124e+00 -2.11602709e+00
  -3.40270880e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -9.02934537e-02  0.00000000e+00
   0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.50000000e+03]
 [ 1.40000000e+01  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  5.00000000e+02]
 [ 1.50000000e+01  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00 -4.72583742e-01 -1.52698955e+00  0.00000000e+00
  -1.60806486e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   4.26712183e-02  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  1.00000000e+03]
 [ 1.60000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  2.00000000e+03]]
```

Figure 16: Phase 2 itr 1

```
tableau for iteration 2:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
   4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
   8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
   1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
   1.60000000e+01  1.70000000e+01]
 [ 0.00000000e+00  0.00000000e+00 -1.58688036e+00 -1.36884424e+00
  -1.92042438e+00  3.70000000e-01  7.80000000e-01  0.00000000e+00
   6.00000000e-01  0.00000000e+00 -8.08126411e-02  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00 -7.40000000e-01
   0.00000000e+00 -7.40000000e+02]
 [ 1.00000000e+00  1.00000000e+00  3.23115124e+00  2.11602709e+00
   3.40270880e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  9.02934537e-02  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 9.00000000e+00  0.00000000e+00  1.33733356e+02  8.04657801e+01
   1.21782214e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.00000000e+00  3.92045147e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 7.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  1.00000000e+03]
 [ 1.30000000e+01  0.00000000e+00 -3.23115124e+00 -2.11602709e+00
  -3.40270880e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -9.02934537e-02  0.00000000e+00
   0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.50000000e+03]
 [ 1.40000000e+01  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  5.00000000e+02]
 [ 1.20000000e+01  0.00000000e+00  0.00000000e+00  2.34350000e+01
   0.00000000e+00 -1.10750000e+01 -3.57850000e+01  0.00000000e+00
  -3.76850000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  2.34350000e+01
   0.00000000e+00  2.34350000e+04]
 [ 1.60000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  2.00000000e+03]]
```

Figure 17: Phase 2 itr 2

```
tableau for iteration 3:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
    4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
    8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
    1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
    1.60000000e+01  1.70000000e+01]
 [ 0.00000000e+00  0.00000000e+00 -2.36688036e+00 -1.36884424e+00
   -1.92042438e+00  3.70000000e-01  0.00000000e+00  0.00000000e+00
    6.00000000e-01  0.00000000e+00 -8.08126411e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -7.80000000e-01 -7.40000000e-01
    0.00000000e+00 -1.13000000e+03]
 [ 1.00000000e+00  1.00000000e+00  3.23115124e+00  2.11602709e+00
    3.40270880e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  9.02934537e-02  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
 [ 9.00000000e+00  0.00000000e+00  1.33733356e+02  8.04657801e+01
    1.21782214e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.00000000e+00  3.92045147e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00]
 [ 7.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
    0.00000000e+00  1.00000000e+03]
 [ 1.30000000e+01  0.00000000e+00 -3.23115124e+00 -2.11602709e+00
   -3.40270880e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00 -9.02934537e-02  0.00000000e+00
    0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  1.50000000e+03]
 [ 6.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
    0.00000000e+00  5.00000000e+02]
 [ 1.20000000e+01  0.00000000e+00  3.57850000e+01  2.34350000e+01
    0.00000000e+00 -1.10750000e+01  0.00000000e+00  0.00000000e+00
   -3.76850000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  0.00000000e+00  3.57850000e+01  2.34350000e+01
    0.00000000e+00  4.13275000e+04]
 [ 1.60000000e+01  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
    1.00000000e+00  2.00000000e+03]]
```

Figure 18: Phase 2 itr 3

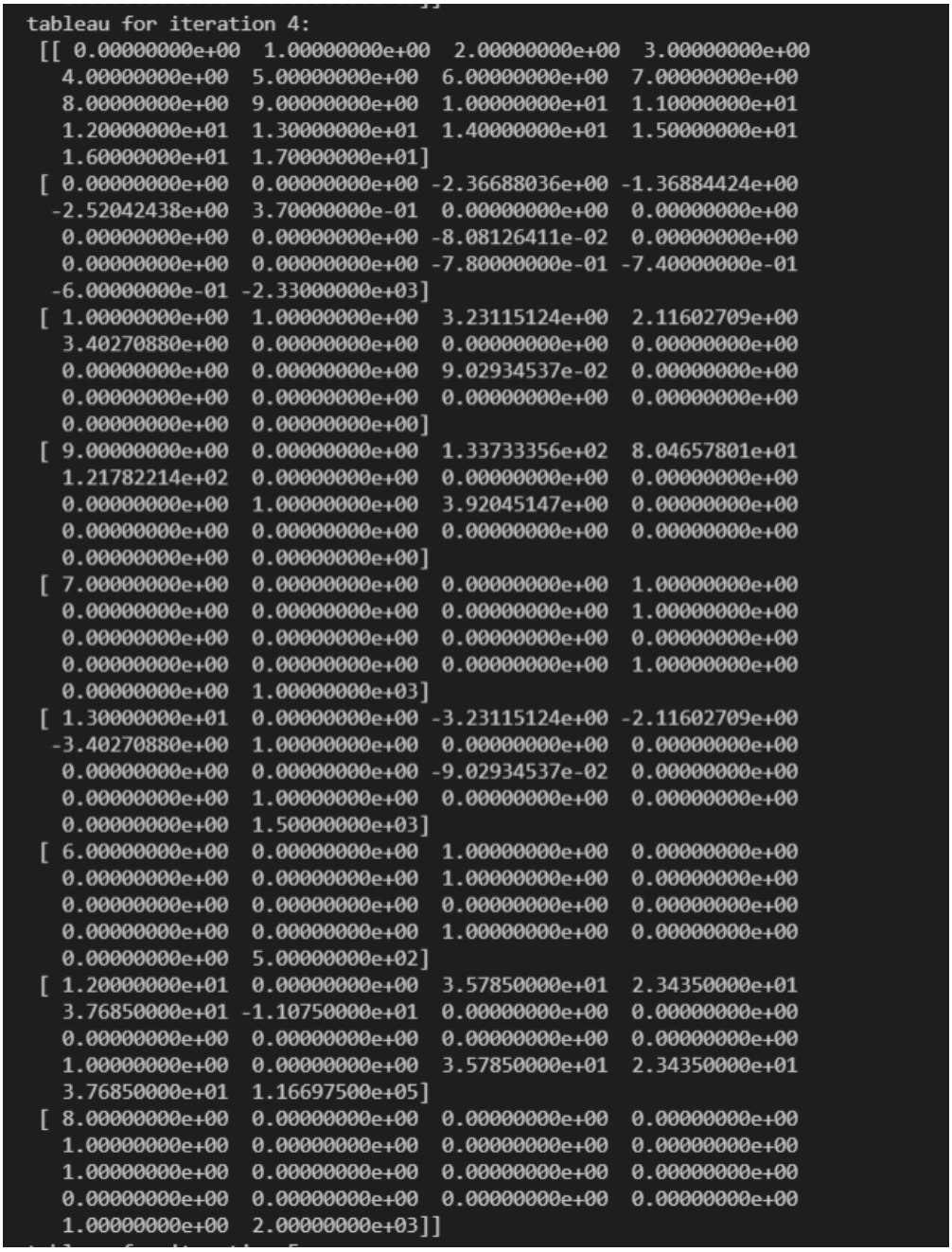

Figure 19: Phase 2 itr 4

```
tableau for iteration 5:
 [[ 0.00000000e+00  1.00000000e+00  2.00000000e+00  3.00000000e+00
   4.00000000e+00  5.00000000e+00  6.00000000e+00  7.00000000e+00
   8.00000000e+00  9.00000000e+00  1.00000000e+01  1.10000000e+01
   1.20000000e+01  1.30000000e+01  1.40000000e+01  1.50000000e+01
   1.60000000e+01  1.70000000e+01]
 [ 0.00000000e+00  0.00000000e+00 -1.17135440e+00 -5.85914221e-01
  -1.26142212e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -4.74040632e-02  0.00000000e+00
   0.00000000e+00 -3.70000000e-01 -7.80000000e-01 -7.40000000e-01
  -6.00000000e-01 -2.88500000e+03]
 [ 1.00000000e+00  1.00000000e+00  3.23115124e+00  2.11602709e+00
   3.40270880e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  9.02934537e-02  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 9.00000000e+00  0.00000000e+00  1.33733356e+02  8.04657801e+01
   1.21782214e+02  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.00000000e+00  3.92045147e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00]
 [ 7.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  1.00000000e+00
   0.00000000e+00  1.00000000e+03]
 [ 5.00000000e+00  0.00000000e+00 -3.23115124e+00 -2.11602709e+00
  -3.40270880e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -9.02934537e-02  0.00000000e+00
   0.00000000e+00  1.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  1.50000000e+03]
 [ 6.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  1.00000000e+00  0.00000000e+00
   0.00000000e+00  5.00000000e+02]
 [ 1.20000000e+01  0.00000000e+00  8.52651283e-14  6.39488462e-14
   1.06581410e-13  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00 -1.00000000e+00  0.00000000e+00
   1.00000000e+00  1.10750000e+01  3.57850000e+01  2.34350000e+01
   3.76850000e+01  1.33310000e+05]
 [ 8.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   0.00000000e+00  0.00000000e+00  0.00000000e+00  0.00000000e+00
   1.00000000e+00  2.00000000e+03]]
```

Figure 20: Phase 2 itr 5

Model 22 tableau Output (Only Phase II shown considering space)

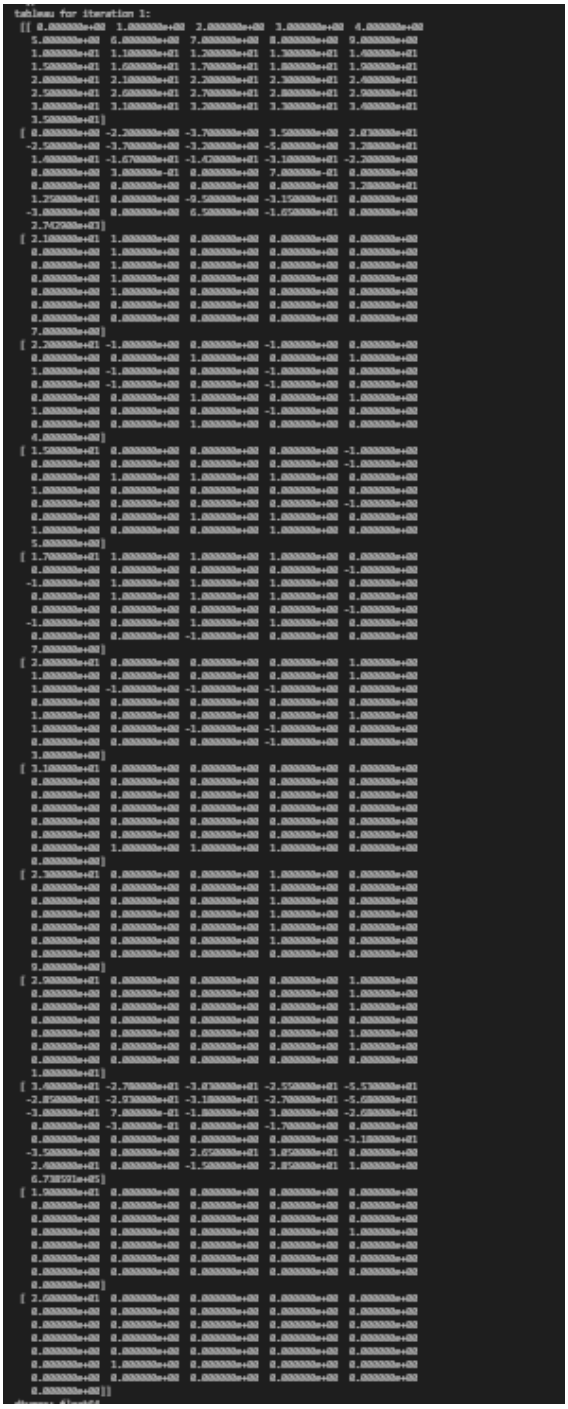

Figure 21: Extra credit Phase 2 itr 1

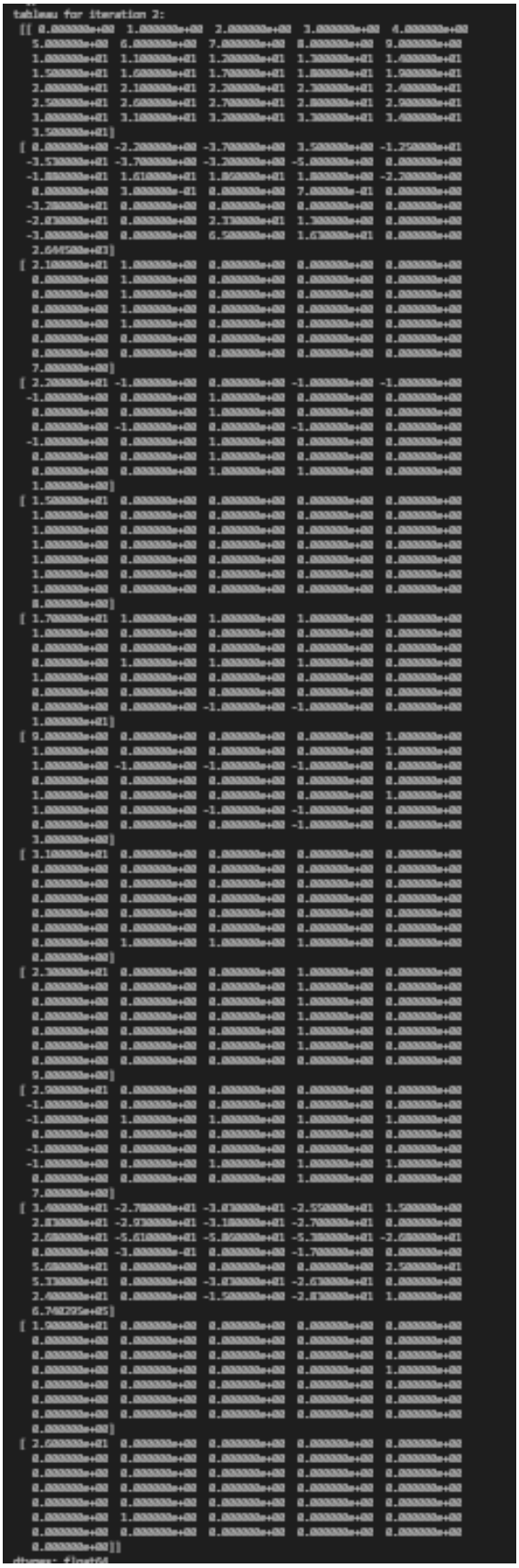

Figure 22: Extra credit Phase 2 itr 2

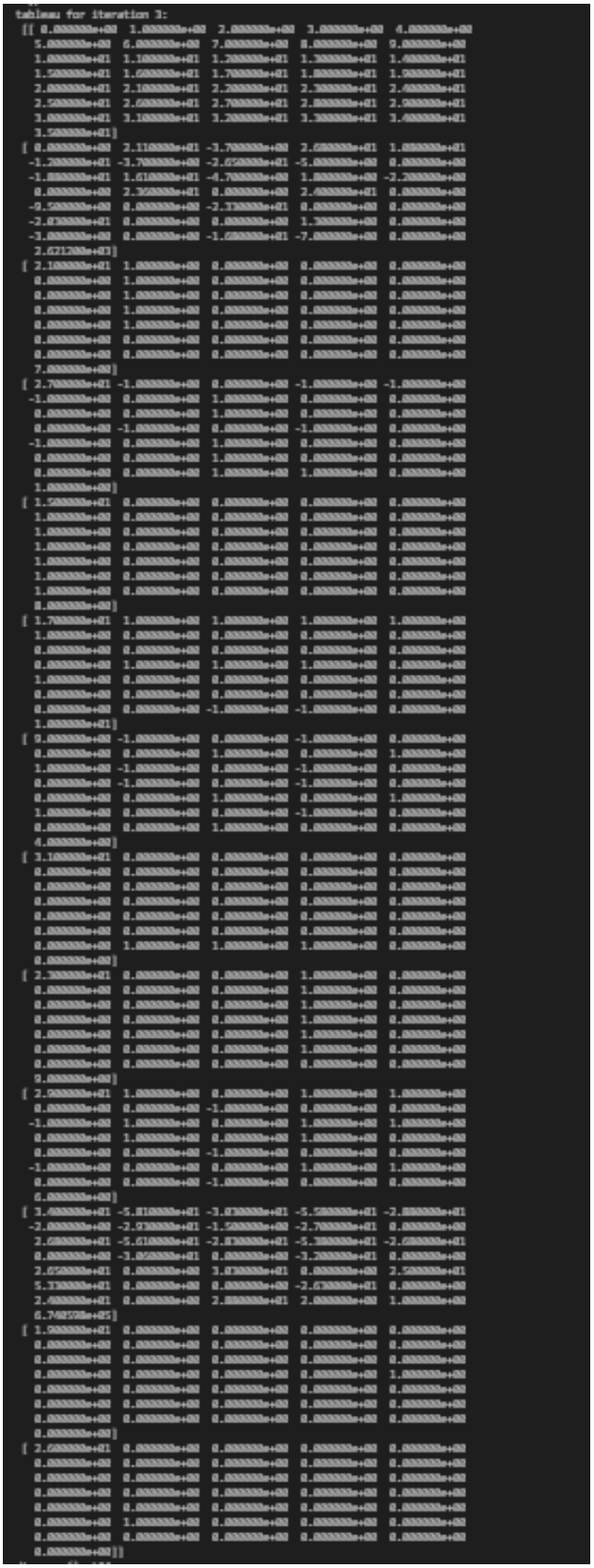

Figure 23: Extra credit Phase 2 itr 3

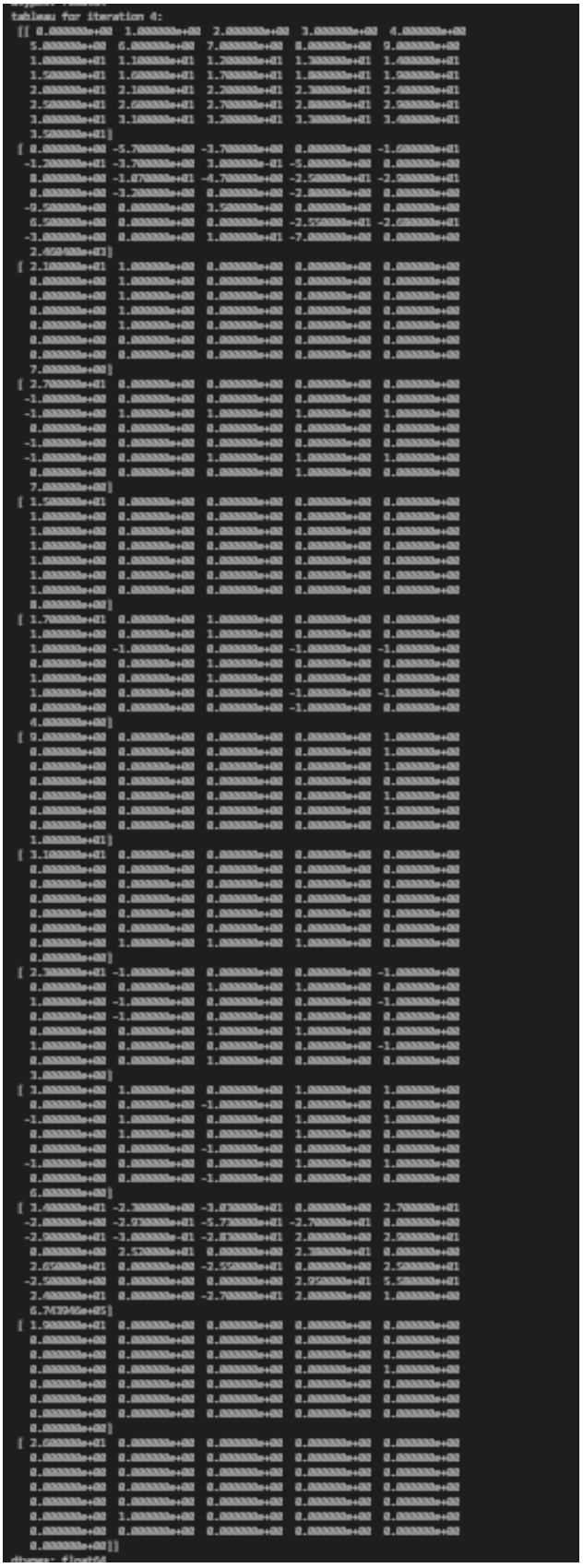

Figure 24: Extra credit Phase 2 itr 4

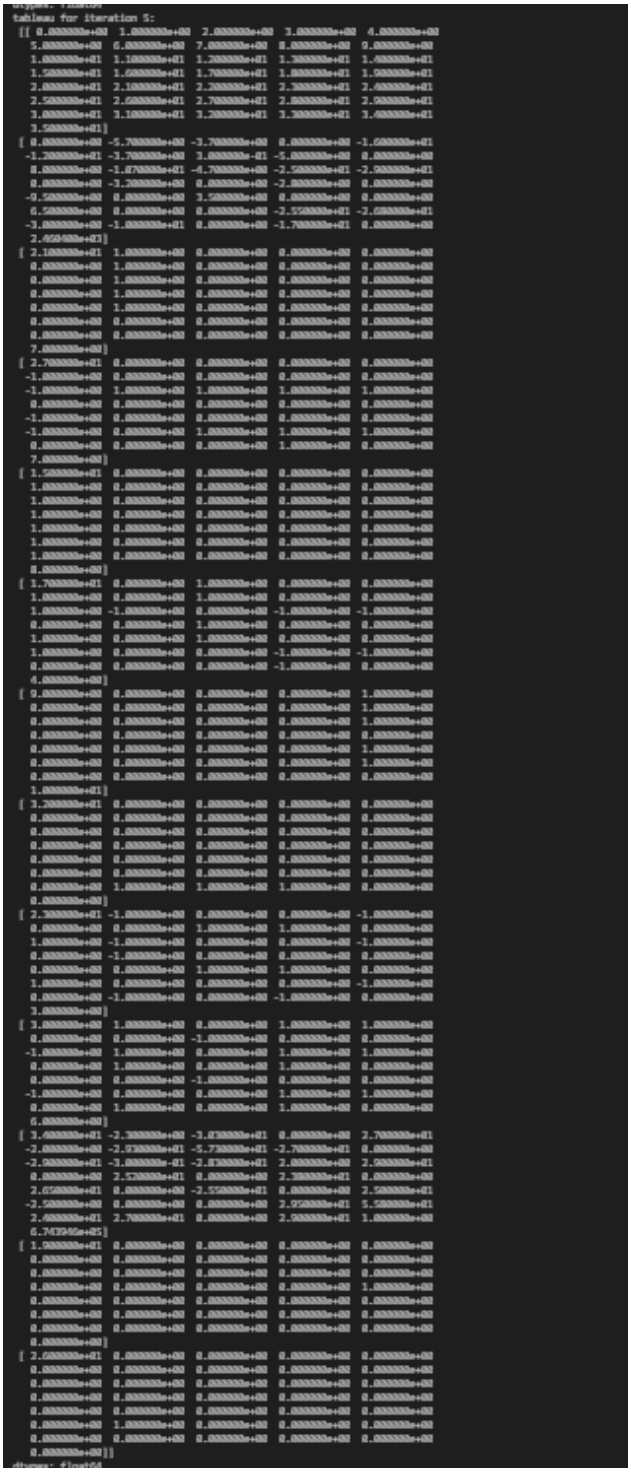

Figure 25: Extra credit Phase 2 itr 5

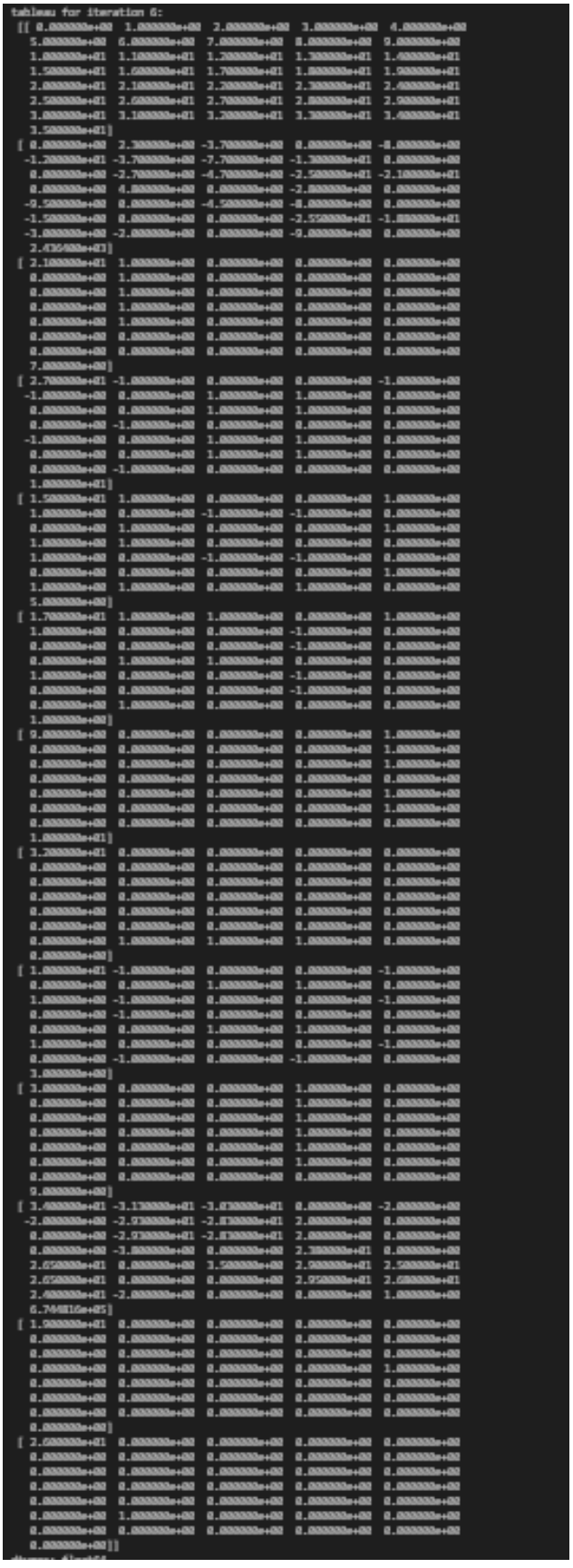

Figure 26: Extra credit Phase 2 itr 6

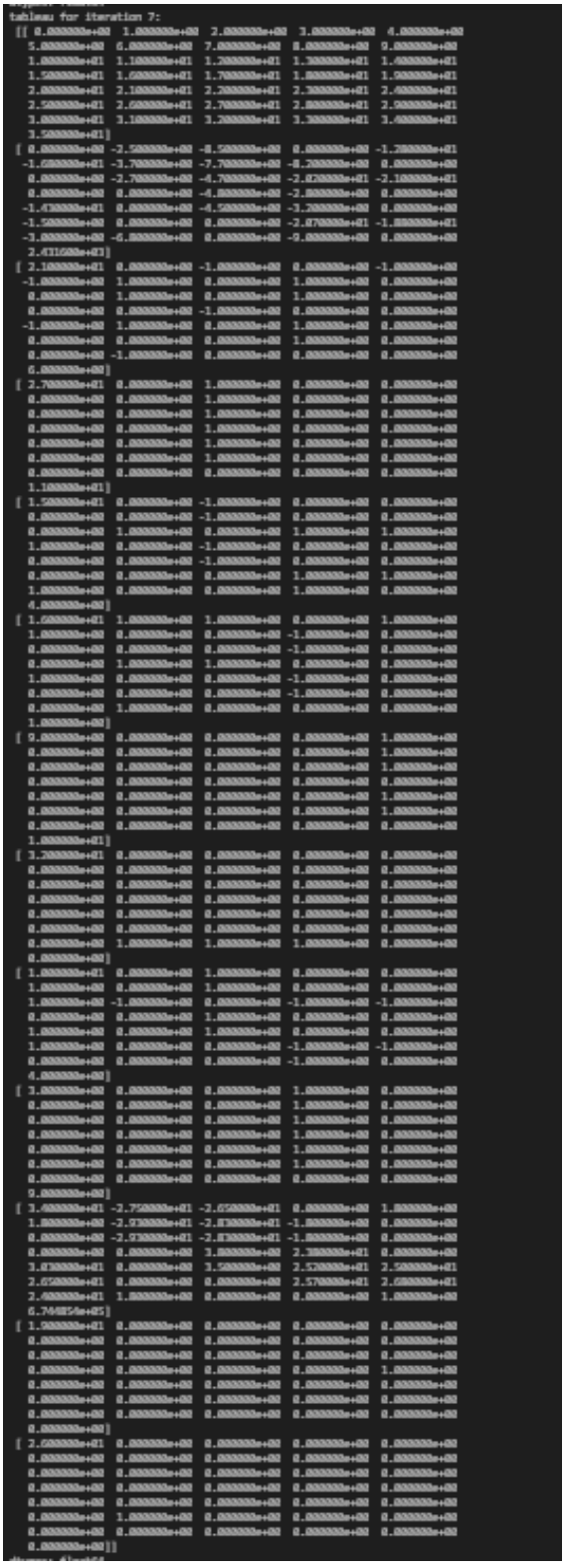

Figure 27: Extra credit Phase 2 itr 7