



---

# Project Proposal for System Engineering: Simplextep - A Python Library for Simplex Method

---

SYSTEM ENGINEERING (20-189)

KEIVAN JAMALI 99104468  
NASSRIN SHARIFI 99104779  
MASTER: DR. MOUSAVI

22/1/2024

## Contents

<b>1</b>	<b>Introduction</b>	<b>2</b>
<b>2</b>	<b>Library Overview</b>	<b>2</b>
2.1	Installation . . . . .	2
2.2	Capabilities . . . . .	2
<b>3</b>	<b>Using Simplextep</b>	<b>3</b>
3.1	Problem Definition . . . . .	3
3.2	Solving with Simplex Method . . . . .	3
3.3	Sensitivity Analysis . . . . .	3
3.4	Solving the Dual Problem . . . . .	4
<b>4</b>	<b>Library Structure and Workflow</b>	<b>5</b>
<b>5</b>	<b>Conclusion</b>	<b>5</b>

### Abstract

This proposal outlines the implementation of the Simplex method in Python, encapsulated in the library "Simplexstep". This library allows users to define and solve linear programming problems using the Simplex method, perform sensitivity analysis, and solve the dual problem. The proposal details the functionality, usage, and structure of the library.

## 1 Introduction

The Simplex method is a widely-used algorithm for solving linear programming problems. The goal of this project is to develop a Python library named "Simplexstep" that simplifies the process of defining and solving such problems. This library also offers features for sensitivity analysis and dual problem solving.

## 2 Library Overview

### 2.1 Installation

To use the Simplexstep library, install it using pip:

```
python -m pip install Simplexstep
```

This library requires the following dependencies but you don't need to install these libraries yourself. It will install them automatically:

- Numpy
- Pandas
- Matplotlib
- Tabulate

### 2.2 Capabilities

The Simplexstep library provides the following functionalities:

- Define and solve linear programming problems using the Simplex method.
- Automatically handle two-phase Simplex if necessary.
- Perform sensitivity analysis.
- Generate and solve the dual problem.

## 3 Using Simplextep

### 3.1 Problem Definition

To define a problem, the library adds slack, artificial, and excess variables as needed to prepare the problem for solving.

```
from Simplextep import ProblemPreparation

# Define your problem here
objective_function = [0, 6, 5, 4]
constraints = [[240, 2, 1, 1], [360, 1, 3, 2], [300, 2, 1, 2]]
equality = ["ineq", "ineq", "ineq"]
parameters = [("x1", "+"), ("x2", "+"), ("x3", "+")]
problem = Problem_Preparation(objective_function=objective_function,
                              constraints=constraints,
                              equality=equality,
                              parameters=parameters,
                              mode="max")
```

Listing 1: Defining a Problem

### 3.2 Solving with Simplex Method

The core of the Simplextep library is solving linear programming problems using the Simplex method.

```
from Simplextep import Simplex

simplex = Simplex(problem)
solution = simplex.fit()
simplex.make_table() # View the steps and previous tables
```

Listing 2: Solving a Problem with Simplex

### 3.3 Sensitivity Analysis

Sensitivity analysis can be performed by analyzing how changes in the right-hand side or objective coefficients affect the solution.

```
from Simplextep import SensitivityAnalysis

analysis = Sensitivity_Analysis(simplex)
analysis.change_righthand(righthands_at_first=[0, 1, 0, 0]) #
    Analyze changes in right-hand side
print(analysis.righthand_nodes) # If needed.
```

Listing 3: Performing Sensitivity Analysis

### 3.4 Solving the Dual Problem

The library also supports generating and solving the dual problem.

```
from Simplextep import Dual

objective_function = [0, 2, 1]
constraints = [[4, 1, 1], [2, 1, -1]]
equality = ["ineq", "ineq"]
parameters = [("x1", "+"), ("x2", "+")]
mode = "max"

dual_problem = Dual(objective_function,
                    constraints,
                    equality,
                    parameters,
                    mode)

dual_problem.fit()
simplex = Simplex(problem=dual_problem.problem)
simplex.fit(max_iterations=10)
```

Listing 4: Solving the Dual Problem

## 4 Library Structure and Workflow

The workflow of using Simplexstep involves the following steps:

1. Define the problem using the `Problem_Preparation` class.
2. Solve the problem using the `Simplex` class.
3. Perform sensitivity analysis using the `Sensitivity_Analysis` class.
4. Solve the dual problem using the `Dual` class.

## 5 Conclusion

The Simplexstep library is a powerful tool for solving linear programming problems using the Simplex method. It provides a comprehensive suite of features including problem definition, solving, sensitivity analysis, and dual problem solving. This proposal highlights the ease of use and functionality of the library, making it accessible for users to perform complex linear programming tasks efficiently.