TECHNICAL SCIENCES

SOFTWARE TOOLS FOR SOLVING LINEAR PROGRAMMING PROBLEMS

Gunay Hasanova
Postgraduate master student
Nakhchivan State University
Nakhchivan, Azerbaijan

Abstract
Linear programming has become one of the most powerful and practical mathematical optimization techniques for decision making in diverse fields such as economics, engineering, transportation, and management science. The continuous growth in computational power and the availability of sophisticated software environments have significantly simplified the formulation and solution of complex LP problems. This paper provides an analytical review of major software tools used for solving LP problems including MATLAB, IBM ILOG CPLEX, Gurobi, LINGO, GLPK, and Python based libraries such as PuLP, SciPy, and Pyomo along with a detailed overview of Excel Solver as an accessible tool for beginners. The article highlights their algorithmic structures, performance characteristics, and real world applicability, concluding with a comparative discussion of their advantages and limitations.

Keywords: Linear programming, simplex method, interior-point method, optimization software, mathematical modeling.

1. Introduction

Optimization plays a crucial role in making rational decisions under constraints. Linear Programming, as a subset of mathematical optimization, involves optimizing a linear objective function subject to a set of linear equality and inequality constraints. Its applications include minimizing cost, maximizing profit, or achieving optimal resource allocation.

Early LP solutions were derived manually using the simplex method, introduced by George Dantzig in 1947. However, the exponential growth in data size and model complexity led to the need for computational solvers capable of handling thousands of variables and constraints efficiently. Today, software tools have become indispensable in operations research, providing automated and accurate solutions through advanced algorithms and numerical optimization techniques.

These software systems not only handle pure LP models but also integrate with mixed integer programming, quadratic programming and nonlinear programming. Their integration with modern languages like python and interfaces such as excel or MATLAB enables researchers, analysts, and engineers to implement optimization directly within broader computational workflows.[2]

2. Theoretical Background of Linear Programming

A general LP problem can be formulated as:

Maximize (or minimize): Z

$$= c_1x_1 + c_2x_2 + \cdots + c_nx_n$$

Subject to:

$$a_{11}x_1 + a_{12}x_2 + \cdots + a_{1n}x_n \le b_1$$
$$a_{21}x_1 + a_{22}x_2 + \cdots + a_{2n}x_n \le b_2$$
$$x_j \ge 0 \forall j = 1,2,\dots,n$$

Here, Z is the objective function, c_j are coefficients, a_{ij} represent constraint coefficients, and b_i are resource limits. The variables $x_j$denote decision variables.

Solution Methods

1. Simplex Method

The simplex method, introduced by George Dantzig (1947), is the most classical and widely used algorithm for solving LP problems. It works by moving along the edges of the feasible region from one vertex to another, improving the objective value at each step until reaching the optimum. Despite its theoretical exponential complexity, the simplex algorithm performs extremely well in practice, solving most small to medium sized problems efficiently. It remains a cornerstone in operations research and is used in resource allocation, production scheduling, and transportation optimization.[1]

2. Dual Simplex Method

The dual simplex method operates similarly to the simplex method but starts with an optimal yet infeasible solution and iteratively restores feasibility while maintaining optimality. It is particularly effective for reoptimization, where only small changes occur in constraints or parameters. This makes it ideal for applications requiring frequent updates, such as rolling production plans or financial portfolio adjustments.

3. Interior-Point Methods

Interior point methods , developed in the 1980s, take a different approach by moving through the interior of the feasible region rather than along its boundaries. They solve a sequence of approximated problems using barrier functions and converge rapidly to the optimal point. These methods are well suited for large scale LP problems with thousands of variables and constraints. IPMs are implemented in advanced solvers like CPLEX and Gurobi, and are used in sectors such as energy optimization, logistics, and telecommunications.[1]

4. Revised Simplex Method

The revised simplex method is a computationally refined version of the standard Simplex algorithm. Instead of storing the full tableau, it maintains only the basis matrix and updates it iteratively, improving numerical accuracy and reducing memory usage. It performs particularly well for sparse and large systems, and most modern solvers, including CPLEX and Gurobi, rely on it for speed and stability.[1-2]

5. Barrier Methods

Barrier methods are a type of interior-point approach that prevent the algorithm from leaving the feasible region by adding logarithmic penalties to the objective function. The barrier parameter is gradually reduced, allowing the solution to approach the optimal boundary. These methods offer excellent numerical stability and are effective for very large or dense problems, often used in combination with the simplex method for final optimization steps.

3. Major Software Tools for Solving LP Problems

MATLAB Optimization Toolbox

MATLAB, developed by MathWorks, is a high level numerical computing environment that provides robust support for linear and nonlinear optimization. The optimization toolbox offers the linprog function to solve LP problems using both the simplex and interior point algorithms.Key features include:

1. Support for dense and sparse matrix operations.
2. Visualization tools for feasible regions and sensitivity analysis.
3. Integration with simulink for control and system modeling.
4. Easy prototyping for research and teaching environments.

MATLAB is commonly used in academia for algorithm development and in industry for process optimization, control systems, and resource planning.[3]

IBM ILOG CPLEX Optimization Studio

IBM CPLEX is a professional-grade solver capable of solving LP, QP, and MILP problems efficiently. It supports multi threaded computation and advanced presolve routines that reduce problem size before optimization.Notable advantages are including:

1. Implementation of simplex, dual simplex, and barrier algorithms.
2. API support for Python, Java, C++, and MATLAB.
3. Built in diagnostics for infeasibility and unboundedness.
4. Highly optimized for enterprise scale problems in logistics, finance, and telecommunication.

CPLEX's performance, stability, and accuracy make it the industry standard for mission-critical optimization applications.[2]

Gurobi Optimizer

Gurobi is one of the fastest commercial solvers, well known for its efficient use of multi core processors. It supports both primal and dual simplex algorithms and advanced interior-point methods.

Strengths include:

1. Automatic model tuning for faster convergence.
2. Built in parameter sensitivity tools.
3. Integration withpPython (via gurobipy), R, and MATLAB.
4. Cloud based license options and academic access.

It is especially favored for machine learning optimization pipelines and high-frequency trading systems, where solution time is crucial.[2]

LINGO and LINDO Systems

LINGO combines a mathematical modeling language and solver in one environment. Its syntax is highly readable and close to algebraic notation.

Distinct advantages:

1. Automatic generation of duals and sensitivity reports.
2. Graphical interface for quick model development.
3. Compatibility with large nonlinear extensions.

LINGO is often used in universities for educational purposes and in small enterprises for project scheduling, production optimization, and resource allocation.[1-2-5]

Microsoft Excel Solver

The solver add-in in Microsoft Excel allows non programmers to model and solve optimization problems using spreadsheet interfaces.

Features include:

1. Support for linear, nonlinear, and integer constraints.
2. Three main algorithms: Simplex LP, GRG Nonlinear, and Evolutionary Solver.
3. Ease of visualization and immediate result interpretation.

Excel solver is particularly suitable for small scale business problems such as production planning, workforce scheduling, and budget allocation. Though limited by problem size, it remains an essential teaching and demonstration tool.[6]

Python-Based Tools for Linear Programming

Python has emerged as a major language for data science and optimization. Its open-source libraries provide flexibility, integration with AI tools, and access to solvers such as CPLEX, Gurobi, or GLPK.

PuLP

PuLP is a modeling interface for LP and MILP problems. It allows users to define decision variables, constraints, and objectives directly in python syntax.[5]

Example:

```
import pulp
x = pulp.LpVariable('x', lowBound=0)
y = pulp.LpVariable('y', lowBound=0)
model = pulp.LpProblem('Example', pulp.LpMaximize)
model += 3*x + 2*y
model += 2*x + y <= 8
model.solve()
```

PuLP can call solvers like CBC, CPLEX, or Gurobi and produce readable reports, making it ideal for integration in analytics workflows.

SciPy.optimize.linprog

SciPy's optimize module provides linprog(), a lightweight tool for LP problems. It supports multiple methods such as Simplex and HiGHs. While not as powerful as Gurobi, SciPy is excellent for scientific computation and quick experiments.[5]

Pyomo

Pyomo is a python based optimization framework that supports linear, nonlinear, and stochastic programming. It can model extremely complex systems and connect to external solvers like GLPK, Gurobi, or CPLEX. It is frequently used in energy systems optimization, water resource management, and supply chain modeling.[5]

Conclusion

Linear Programming remains an essential technique for optimization in both academia and industry. The evolution from manual calculations to advanced computational solvers has revolutionized how complex decisions are made. Tools such as Gurobi, CPLEX, and MATLAB set the benchmark for industrial-grade performance, while open-source packages like GLPK, PuLP, and Pyomo democratize access to optimization technologies. For smaller scale or educational use, excel solver continues to serve as a valuable starting point.The future of LP software development lies in its integration with artificial intelligence, cloud based optimization, and big data analytics, making optimization more adaptive, intelligent, and accessible to all users.

References

1. Winston, W. L. Operations Research: Applications and Algorithms. Cengage Learning
2. IBM Corporation. (2024). IBM ILOG CPLEX Optimization Studio Documentation.
3. Gurobi Optimization LLC. (2024). Gurobi Optimizer Reference Manual.
4. The MathWorks, Inc. (2023). MATLAB Optimization Toolbox User’s Guide.
5. Makhorin, A. (2023). GNU Linear Programming Kit (GLPK) User Manual. Free Software Foundation.
6. Mitchell, S., & Hart, W. E. (2020)Pyomo – Optimization Modeling in Python. Springer.
7. Microsoft Corporation. (2023). Excel Solver Reference Guide.