Mathematical Framework to Compute Optimal x_i at Time t_i

We are given:

- A total order size S that must be completed by the end of day.
- The day is split into N discrete trading intervals.
- For each time step t_i, we have a temporary impact function $g_t(x) \approx \beta_t x$, assumed linear.

Goal:

Determine the allocation vector $x = (x_1, x_2, ..., x_N)$ such that:

- The total shares traded satisfies: $\sum x_i = S$
- The total temporary impact (slippage) cost: $\sum \beta_i * x_i$ is minimized.

Mathematical Formulation (Linear Program):

We set this up as a constrained linear optimization problem:

Minimize: $\sum \beta_i * x_i$ Subject to: $\sum x_i = S$ $x_i \geq 0$ for all $i \in \{1, ..., N\}$

Interpretation:

- β_i: Cost per share at time t_i (i.e., slope of the linear temporary impact model at t_i).
- x_i: Number of shares to execute at time t_i.

Since both the objective function and constraints are linear, this problem can be efficiently solved using standard Linear Programming (LP) solvers such as:

- scipy.optimize.linprog (Python)
- Gurobi, CPLEX, or other commercial solvers

Algorithm Sketch:

1. **Input**: S, vector $\beta=(\beta_1,\beta_2,\ldots,\beta_N)$
2. **Define LP**:
   - Objective: $\min \beta^T x$
   - Constraints:
     - $Ax=S$ where $A=[1\ 1\ \ldots\ 1]\in R^{1\times N}$
     - $x\geq 0$
3. **Solve** LP using simplex or interior-point method
4. **Output**: Optimal allocation x^*

1.

Notes:

- The LP ensures the solution is globally optimal due to convexity.
- If integer quantities are required, this becomes an Integer Linear Program (ILP), solvable using branch-and-bound methods.
- If future β_i values are unknown at time t_i, then a model predictive control (MPC) or online optimization variant would be necessary.