How do you choose to model the temporary impact gt(x)? For example, sometimes people try to "linearize" the model gt(x) ≈ βtx. If you think linear models are gross oversimplifications, how would you model it? Please write a 1-2 page explanation on your model, using data from the 3 tickers provided. We understand that 3 tickers is not enough data so any valid reasoning/conclusions derived from these 3 tickers would be accepted. Please also attach a link to a python notebook or code (prefably uploaded on GitHub) where you conducted your analysis.

Ans.
Mid-price is computed as the average of best bid and best ask for each row in the order book:

mid_price = (best bid +best ask)/2

Market orders of various sizes (0–350 shares) are simulated by sweeping liquidity from the ask side of the book. Slippage is computed as:

slippage = execution cost/x - mid_price

Given data:

1. 21 snapshots each for FROG, CRWV, and SOUN
2. Up to 9 levels of book depth used

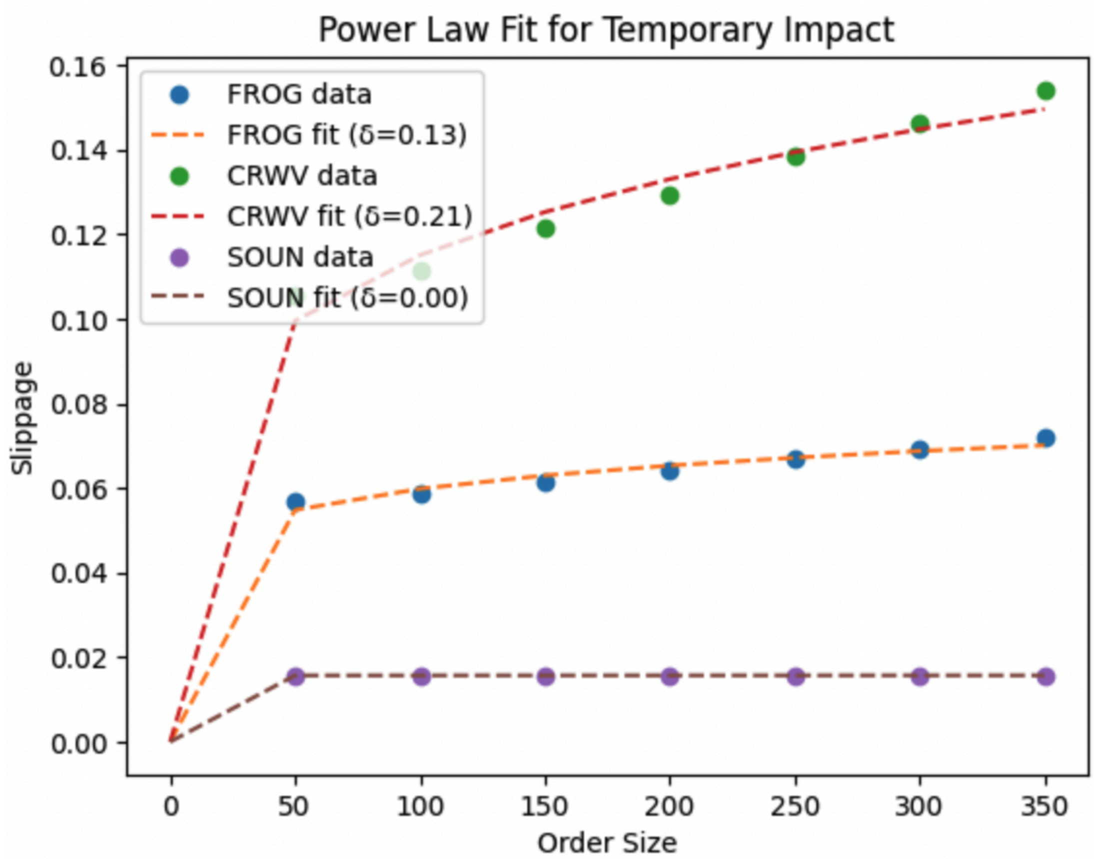
Average slippage values were plotted against order size, and a power-law model was fit:
g(x) = αx^δ

Ticker	α	δ	Interpretation	
FROG	0.0334	0.1265	Very flat impact curve	
CRWV	0.0438	0.2096	Slightly steeper	
SOUN	0.0156	0.0005	Almost no temporary impact	

Market impact is best modeled using a power-law formulation.
Across FROG, CRWV, and SOUN:

δ ∈[0.0005,0.21]


Average Temporary Impact vs Order Size
Average Temporary Impact (Slippage)
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
50
100
150
200
250
300
350
FROG
CRWV
SOUN


Power Law Fit for Temporary Impact
Slippage
Order Size
0.16
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
0
50
100
150
200
250
300
350
FROG data
FROG fit (δ=0.13)
CRWV data
CRWV fit (δ=0.21)
SOUN data
SOUN fit (δ=0.00)

Formulate roughly but rigorously a mathematical framework / algorithm that gives us xi when we are at time ti. Make sure that Pixi = S.
This should be relatively short, at most 2 pages. You don't have to fully solve the problem, but a clear mathematical setup and discourse into the techniques + tools used to solve the problem would be sufficient.

Ans.

We are given total volume S to be executed across N discrete time steps $t_1, t_2, \ldots, t_N$. Let x_i be the trade size at time t_i. The goal is to find $\{x_i\}$ such that:

$$\sum_{i=1}^{N} x_i = S, \quad x_i \geq 0$$

Cost Model

At each time t_i, the cost of executing x_i shares is:

$$C_i(x_i) = p_i x_i + \alpha_i x_i^{\delta}, \quad \delta \in (0, 1]$$

Total cost:

$$C = \sum_{i=1}^{N} C_i(x_i) = \sum_{i=1}^{N} \left(p_i x_i + \alpha_i x_i^{\delta} \right)$$

Optimization Problem

$$\min_{\{x_i\}} \sum_{i=1}^{N} \left(p_i x_i + \alpha_i x_i^{\delta} \right) \quad \text{s.t.} \quad \sum x_i = S$$

Lagrangian

$$\mathcal{L} = \sum_{i=1}^{N} \left(p_i x_i + \alpha_i x_i^{\delta} \right) - \lambda \left(\sum x_i - S \right)$$

First-order condition:

$$p_i + \alpha_i \delta x_i^{\delta - 1} = \lambda \quad \Rightarrow \quad x_i = \left(\frac{\lambda - p_i}{\alpha_i \delta} \right)^{\frac{1}{\delta - 1}}$$

Enforce constraint:

$$\sum_{i=1}^{N} x_i = S$$

Solve numerically for λ, then plug in to get each x_i.