Abhishek R-P - Markov chains in options

quickly explain Markov chains to me how they are used in the tech products and then um tell me if they can be used in backtesting options strategies like can can they be tested on options strategy in terms of momentum scalping kind of a deal on bank nifty options and nifty options in indian stock market some strategies around that that you believe can be highly profitable you can use stop loss trailing stop loss etc and as needed and you can include variables like volume or whatever but yeah give me some really high quality strategies

Got it 👍 let's do this step by step:

🔹 What's a Markov Chain?

- A **Markov Chain** is a mathematical model for a system that moves from one state to another, where the **probability of the next state depends only on the current state** (not the full history).
- Example: Weather → if today is sunny, tomorrow has 80% chance sunny, 20% rainy. Yesterday doesn't matter.

In tech products:

- **Google PageRank** (early Google used a Markov model of a "random surfer").
- **Recommendation systems** (probability you move from one type of video/content to another).
- **Predictive text / NLP** (older models before deep learning).
- **Game AI / Reinforcement learning** (modeling states → transitions).
- **Finance models** (regime switching: trending vs mean-reverting).

🔹 Can Markov Chains be used in Options Backtesting?

Yes, but with nuance:

- Options scalping momentum relies on **micro price dynamics** (IV spikes, OI shifts, tick data).
- Markov Chains can model **state-to-state transitions**, e.g.:

    - State A: price trending up + IV rising + volume surge.
    - State B: sideways + low volume.
    - State C: reversal down.
- You estimate transition probabilities between states → simulate future scenarios → backtest strategy rules.

They're not perfect because markets aren't strictly memoryless, but they can **simplify regime modeling**. For intraday scalping (Bank Nifty/Nifty), it's useful for **state probabilities** rather than raw price prediction.

🔹 Strategy Ideas Using Markov + Momentum Scalping

Here are **practical, testable strategies** that can be backtested:

---

1. Momentum Burst State Model

- **States:**
    - S1: Price > EMA5 > EMA20, volume > 1.5× avg (bullish burst).
    - S2: Price < EMA5 < EMA20, volume > 1.5× avg (bearish burst).
    - S3: Neutral/sideways (low volume).
- **Transitions:**
  Estimate probabilities: e.g. S1 → stays S1 (70%), S1 → S3 (20%), S1 → S2 (10%).
- **Trade Logic:**
    - Enter CE scalp if S1 persists ⩾ 2 candles.
    - Enter PE scalp if S2 persists ⩾ 2 candles.
    - Exit if transition probability < 50% (detected reversal risk).
- **Risk mgmt:**
    - SL = 0.5× ATR
    - Trailing SL = EMA5 breach.

---

2. Volatility Spike Capture (Markov + IV/Volume)

- **States:**
    - Low IV, low OI change (calm).
    - IV spike + OI up (new trend forming).
    - IV spike + OI down (short covering).

- **Trade Logic:**
  - When IV spike + OI up → scalp CE/PE in trend direction.
  - When IV spike + OI down → scalp opposite (fade).
- **Exit:**
  Use trailing SL at VWAP ±0.3%.
- **Why profitable?**
  Captures **institutional flows** → these drive Bank Nifty scalps.

---

3. Transition Probability Breakout

- Instead of fixed thresholds, compute real-time **transition probabilities**:
  - If P(Sideways → TrendUp) > 65% → buy CE.
  - If P(Sideways → TrendDown) > 65% → buy PE.
- **Indicators for probability calc:**
  - Volume Delta (Buy volume vs sell volume).
  - Order Book Imbalance (best bid/ask).
  - Price near intraday high/low.

---

4. Hybrid Markov + Stop-loss Grid

- Trade 1–2 lots.
- Enter on momentum states.
- SL = 20 points (Bank Nifty), trail 15 points.
- **Grid scaling:** If state persists for 3 transitions in a row → add another lot (compounding momentum).
- **Exit all** on first opposite state confirmation.

🔹 Why This Can Work in India (Bank Nifty/Nifty Options)

- **High liquidity** → momentum bursts happen often.
- **OI + volume data** → gives cleaner state definitions.
- **Scalping environment** → Markov chains capture short memory → ideal fit.

✅ Next Step if you want to actually test:

- Define states programmatically (EMA, volume, OI thresholds).

- Compute transition matrices from historical tick/1-sec data.
- Backtest trade rules with slippage + brokerage.
- Optimize for expectancy (avg profit × win rate − avg loss × loss rate).

Do you want me to actually **sketch a backtest framework in Python** (with pseudocode for Bank Nifty options states + trades), so you could plug in your data and run it? That way you'll see if the Markov-based scalping edge is real.