Specification of the Dealer Module for the Kalecki Ring (Bilancio Simulation)

Contents

1 Purpose and Scope

This document specifies how to implement a one–security dealer module, based on the L1 baseline dealer kernel, inside the existing Kalecki debt ring simulation in `Bilancio`. The dealer makes a two–sided market in secondary claims on ring debts and provides liquidity to agents who:

- sell debt tickets to raise cash to meet their own maturing obligations;
- buy debt tickets as an investment to realize capital gains later.

The implementation uses:

- a fixed standard ticket size S (typically $S = 1$);
- maturity buckets (e.g. short/medium/long), each with its own dealer instance;
- sequential, one–ticket customer flow ("Option A"): one trade at a time, with quotes as a function of current inventory only.

2 Limitations and Modeling Scope

The dealer module for the Kalecki ring is intentionally minimal. This subsection summarizes key limitations and scope choices built into the current specification.

Backward–looking, bucket–level VBT anchors. Value–based trader anchors (M_t^b, O_t^b) for each bucket b are updated only from *realized* losses on tickets that were in bucket b and matured in the previous period, via the loss rate ℓ_{t-1}^b and the linear rules

$$M_t^b = M_{t-1}^b - \phi_M^b \, \ell_{t-1}^b,$$
$$O_t^b = O_{t-1}^b + \phi_O^b \, \ell_{t-1}^b.$$

There is no dependence on current–period order flow, dealer inventory, or cross–bucket spillovers. In particular:

- VBT mids and spreads are purely backward–looking and react only to defaults in the immediately preceding period.
- Losses in one bucket do not directly affect VBT anchors in other buckets.

This keeps the anchor process simple but omits richer term–structure and contagion effects.

Continuous prices and absence of tick size. Dealer and VBT quotes $(A_t^b, B_t^b, a_c^b(x), b_c^b(x))$ are specified in continuous price space. The specification does not impose a discrete tick size or round quotes to a grid. In practical implementations one may wish to introduce a tick $\Delta > 0$ and define all prices on the lattice $\Delta\mathbb{Z}$, but this is outside the baseline model.

Single–issuer asset exposure for traders. On the asset side, each ordinary (non–dealer, non–VBT) trader i may hold debt claims on at most one issuer $k(i)$, while their own liability is owed to a single creditor $j(i)$ at a single maturity. This "one–issuer on each side" structure greatly simplifies bookkeeping and interpretation of defaults, but it rules out portfolio diversification and more complex exposure patterns at the trader level.

No interest accrual and no coupon structure. All tickets are zero–coupon claims of fixed face value S with a single payoff at maturity. There is no explicit term structure of interest rates, no coupon schedule, and no discounting; pricing and equity measurement are expressed in levels (face and VBT mids) rather than yields. This is adequate for tracking cash/default dynamics but cannot represent interest–rate risk or coupon timing.

Restricted market architecture. The trading architecture is deliberately constrained:

- All secondary trading is intermediated by the bucket dealer; there are no direct trader–trader or trader–VBT trades.
- Dealers and VBTs hold no debt liabilities and do not borrow; their balance sheets consist only of cash and bucket–eligible tickets, with equity as the residual.
- There is no new debt issuance during the simulation; all liabilities are specified and ticketized at initialization.

These restrictions isolate the interaction of cash flows, defaults, and the dealer/VBT pricing mechanism, but exclude leverage dynamics, bilateral OTC markets, and primary market activity.

Random order flow with hard per–period limits. Customer arrivals to the dealer are modeled as sequential, one–ticket events governed by a Bernoulli side selector with parameter π_{sell} and a per–period cap $N_{\max}$ on arrivals. An agent's eligibility to BUY or SELL is determined by simple liquidity and investment rules, and once an agent trades on a given side in period t it is removed from the corresponding eligibility set. This design produces a tractable, state–dependent but stylized order–flow process; it does not aim to reproduce realistic high–frequency microstructure.

Scope of potential extensions. The above limitations are by design rather than oversights. Natural extensions include: (i) adding flow– and inventory–sensitive components to the VBT update rule, (ii) introducing price ticks and discrete grid constraints, (iii) relaxing the single–issuer exposure restriction for traders, (iv) allowing interest accrual and coupon payments, and (v) enriching the market architecture with limited trader–trader or trader–VBT trading and controlled dealer/VBT leverage. These are left for future versions of the module.

3 Simulation Event Loop at a Single Time Point

At each time point t, the system executes the following stages in order. Only after all stages are completed does the simulation advance to time $t + 1$.

1. **Update maturities in buckets.**
   - Update all maturity buckets and identify which balance sheet entries (assets and liabilities) mature at time t.
   - Mark these entries as due for potential settlement later in the loop.
2. **Dealer quotation stage.**
   - Given his current balance sheet (cash, inventories, limits, etc.), the dealer computes and posts quotes for all relevant instruments.
   - These quotes are the terms at which agents may trade with the dealer during time t.
3. **Agent liquidity assessment.**
   - Each agent computes its liquidity shortfall at time t:
     - Determine current payment obligations (from maturing liabilities and other due payments).
     - Compare required payments to available cash (and other immediately usable means of payment).
   - For each agent, classify its situation:

- *Shortfall-facing*: has due payments that exceed available cash.
- *No shortfall / surplus*: has no unpaid due obligations and may have surplus cash.

4. **Eligibility to trade with the dealer.**
   - *Eligibility to sell*: an agent is eligible to sell if
     (a) it is shortfall-facing at time t, and
     (b) it holds some asset (a claim on another agent) that can be offered to the dealer.
   - *Eligibility to buy*: an agent is eligible to buy if
     (a) it has no liquidity shortfall at time t, and
     (b) it holds surplus cash that it is willing to deploy into assets.
   - Construct two sets: the set of eligible sellers and the set of eligible buyers.
5. **Dealer interaction sub-loop.**
   - While there remain agents in either the eligible-seller set or the eligible-buyer set:
     (a) Randomly draw one eligible agent.
     (b) Determine whether this agent acts as a buyer or seller (given its classification and current state).
     (c) Route the agent to the appropriate dealer for the relevant instrument.
     (d) Execute the trade at the prevailing dealer quotes:
       - Update the balance sheet of the agent (cash and asset/liability positions).
       - Update the balance sheet of the dealer (cash and inventory).
     (e) Record the transaction.
     (f) Re-check the agent's liquidity status; if it no longer wants or needs to trade, remove it from the eligible set(s).
   - The sub-loop terminates when all agents that wish to buy or sell at time t have been exhausted.
6. **Settlement of maturing debt.**
   - Execute settlement for all balance sheet entries that mature at time t:
     (a) For each maturing liability, attempt payment in the designated means of payment.
     (b) If the liable agent has sufficient cash, transfer cash to the asset-holding counterparty and extinguish the corresponding asset and liability.
     (c) If the liable agent does not have sufficient cash, trigger a default event for that agent.
7. **Default resolution and loss distribution.**
   - If a default occurs at time t:
     (a) Record the default event (for statistics and analysis).
     (b) Distribute the remaining cash on the balance sheet of the defaulting agent to its liability holders according to the predefined allocation rule.
     (c) Adjust the corresponding asset and liability entries on the balance sheets of all affected counterparties.
   - The simulation does *not* stop at default; it continues with adjusted balance sheets.
8. **Post-default re-quotation (VBT and dealer).**
   - If any defaults occurred at time t:
     (a) The value-based trader (VBT) recomputes and updates their outside quotes according to the specified rule (taking into account defaults and balance sheet changes).
     (b) Given the new outside quotes and his own updated balance sheet, the dealer recomputes and updates his quotes.
9. **End-of-day and transition to next time point.**
   - After quotation updates, all balance sheets and quotes at time t are considered finalized.
   - The system advances to the next time point $t + 1$, where the same sequence of stages

is repeated.

4 Environment: Kalecki Debt Ring + Tickets

4.1 Ring debts and ticketization

- There are N ring agents indexed by $i = 1, \dots, N$.
- The original Kalecki ring specifies nominal liabilities L_{ij} from agent i to agent j with face amounts and due dates (discrete time periods).
- Fix a standard ticket size $S > 0$ in units of face value (recommended: $S = 1$).

Ticketization. Each liability L_{ij} with face amount A_{ij} and maturity τ_{ij} is represented as a collection of tickets:

- Number of tickets: $n_{ij} = A_{ij}/S$ (assume divisibility or pre–round upstream).
- For each ticket $k = 1, \dots, n_{ij}$, create a record:
  - `issuer`: i (the debtor);
  - `owner`: initial creditor j;
  - `face`: S;
  - `maturity_time`: τ_{ij} (discrete period index);
  - `bucket_id`: derived from remaining time to maturity (see below).

4.2 Maturity buckets

Define a small number of maturity buckets (recommended: three) with disjoint ranges in remaining time–to–maturity τ:

- Short: $\tau \in \{1, 2, 3\}$;
- Medium: $\tau \in \{4, \dots, 8\}$;
- Long: $\tau \geq 9$.

At each simulation period t, recompute for every ticket:

- `remaining_tau` $=$ `maturity_time` $- t$;
- `bucket_id` $=$ function of `remaining_tau` (short/medium/long).

Each bucket has its own dealer instance (and its own outside VBT anchor), but tickets can move between buckets as τ shrinks over time.

5 Actors and State

5.1 Ring agents

For each agent i we need, at each period t:

- `cash_i(t)`: current means of payment;
- `tickets_owned_i(t)`: list of ticket IDs currently owned by i;
- `obligations_due_i(t)`: list of tickets they must pay at t (issuer $= i$, maturity_time $= t$).

All balance–sheet updates (cash and ticket ownership) must be done in a double–entry manner.

5.2 Value–based trader (VBT) per bucket

For each bucket b, define exogenous outside anchors:

- Midpoint M_t^b;
- Spread O_t^b;
- Outside quotes:

$$A_t^b = M_t^b + \frac{O_t^b}{2}, \qquad B_t^b = M_t^b - \frac{O_t^b}{2}.$$

In the baseline implementation, M_t^b and O_t^b are given exogenously or from a simple rule (e.g. fixed over time or slowly adjusted based on default rates). The VBT is assumed to have effectively infinite depth at these prices. In the baseline implementation, (M_t^b, O_t^b) are updated by the loss-based rule in Section 10 unless explicitly held fixed for an experiment.

5.3 Dealer per bucket

For each bucket b, the dealer state at period t consists of:

- x_t^b: inventory of bucket–b tickets on the dealer's shelf;
- a_t^b: securities balance (for bookkeeping, typically $a_t^b = x_t^b/S$);
- C_t^b: dealer cash in common units of account;
- derived: equity at VBT mid $E_t^b = C_t^b + M_t^b\, a_t^b \quad (\text{with } a_t^b = x_t^b/S)$.

Accounting convention (equity bases). Traders measure both assets and liabilities at face values. Dealers and VBTs measure their security inventories at the VBT mid M_t^b for the corresponding bucket (equity $E = C + Mx$). This asymmetry is intentional: traders are modeled as booked at par, while market makers are marked to the prevailing outside mid.

6 Balance sheet Set Up Rules

6.1 Population Size and Market-Maker Configuration

The baseline configuration of the simulation imposes the following constraints on the number and types of agents:

- The total number of **trading agents** (i.e. ordinary non-dealer, non-VBT entities that can hold cash and debt claims, and issue debt) is *at least* 100. In other words,

  $$N_{\text{traders}} \geq 100.$$

  The simulator should not be initialized with fewer than 100 such agents.
- There are *exactly three* **dealers** in the system:

  $$N_{\text{dealers}} = 3.$$

  Each dealer is associated with one of the three maturity buckets (short, medium, long) and makes a market only in the tickets belonging to its own bucket.
- There are *exactly three* **value–based traders (VBTs)**:

  $$N_{\text{VBT}} = 3.$$

  Each VBT is likewise associated with one of the three maturity buckets and provides the outside quotes (A, B) for that bucket.

Thus, the overall agent population decomposes as

$$N_{\text{total}} = N_{\text{traders}} + N_{\text{dealers}} + N_{\text{VBT}},$$

with the hard constraints

$$N_{\text{traders}} \geq 100, \qquad N_{\text{dealers}} = 3, \qquad N_{\text{VBT}} = 3.$$

No additional dealers or VBTs may be introduced, and the number of dealers/VBTs must not vary over the course of the simulation.

6.2 Balance-Sheet Structure and Admissible Positions

This section specifies the admissible balance-sheet structure for all entities in the simulation. The aim is to keep the financial architecture minimal and transparent, so that all complexity arises from the interaction of cash, debt claims, and the dealer/VBT pricing mechanism.

Common structure: assets and liabilities

All entities in the simulation share the following structural constraints:

- On the **asset side**, every entity may hold only:
  - cash (the settlement asset / means of payment);
  - debt claims (tickets) on other entities (where traders only hold claims on one other entity, as specified below, and dealers/VBTs can hold claims on multiple entities as long as those claims are of the right maturity bucket).
- No entity holds any other type of financial or real asset: there are no equities, no physical assets, and no derivatives in the baseline specification.

On the liability side, entities are differentiated as follows.

Traders (non-dealer, non-VBT agents)

By "traders" we mean all ordinary agents in the simulation that are neither dealers nor value–based traders. Their balance sheets obey:

- **Assets:**
  - cash holdings $C_i \geq 0$;
  - a debt claims (tickets) issued by *one single other trader*.
- **Liabilities:**
  - each trader i has a *single* debt liability in the baseline model:
    * this liability is owed to exactly *one* counterparty, denoted $j(i)$, which can be another trader, a dealer, or a VBT;
    * the entire nominal amount of trader i's debt has a *single maturity date* T_i;
    * when ticketized, the liability of trader i is split into n_i tickets of size S, but all such tickets share the same maturity T_i and the same creditor $j(i)$.
- **Exposure constraint (asset side):**
  - on the asset side, in the basic configuration we impose a *single–counterparty* restriction for simplicity:

    for each trader i, all debt claims held by i are issued by the same counterparty $k(i)$.

The net worth (equity) of trader i is then defined as the residual:

$$E_i = C_i + \sum_{\ell \in \mathcal{A}(i)} \text{face\_value}(\ell) - \sum_{m \in \mathcal{L}(i)} \text{face\_value}(m),$$

where $\mathcal{A}(i)$ is the set of tickets held by i and $\mathcal{L}(i)$ is the set of tickets issued by i (all with the same maturity T_i in the baseline).

Dealers

Dealers are specialized market–making agents with the following balance-sheet restrictions:

- **Assets:**
  - cash holdings $C^{\mathrm{D}} \geq 0$;
  - positions in the traded security (tickets) for their assigned bucket(s). These positions are exactly the inventory x used in the pricing kernel.
- **Liabilities:**

– dealers hold *no explicit debt liabilities* in the simulation:

$$\mathcal{L}^{\mathrm{D}} = \varnothing.$$

– the dealer's net worth (equity) is a pure residual:

$$E^{\mathrm{D}} = C^{\mathrm{D}} + Mx,$$

where M is the relevant VBT mid for the security being made.

Thus dealers do not borrow in the model; they are constrained to trade using their own cash and inventory only, in line with the no–borrowing and no–shorting conditions.

Value–Based Traders (VBTs)

Value–based traders are outside anchor agents that price credit and provide effectively infinite depth at the outside quotes (A, B). Their balance sheets are constrained as follows:

- **Assets:**
  - cash holdings $C^{\mathrm{VBT}} \geq 0$;
  - positions in the same security as the dealer in the corresponding bucket, used to conceptualize their role as long–term holders.
- **Liabilities:**
  - VBTs hold *no explicit debt liabilities*:

$$\mathcal{L}^{\mathrm{VBT}} = \varnothing.$$

  - their equity is again the residual:

$$E^{\mathrm{VBT}} = C^{\mathrm{VBT}} + Mx^{\mathrm{VBT}}.$$

Operationally, the VBT's internal balance sheet does not constrain its quoting; it serves primarily for consistency of accounting and for diagnostics (e.g. tracking how many tickets have been absorbed at the outside quotes).

Summary of restrictions

In the baseline version of the simulator, we therefore impose:

- All entities (traders, dealers, VBTs) hold only *cash* and *debt claims* as assets.
- Traders:
  - have exactly one debt liability, owed to a single counterparty, with a single maturity date (ticketized if needed);
  - may hold debt claims on at most one issuer on the asset side in the baseline.
- Dealers and VBTs:
  - hold *no* debt liabilities;
  - have balance sheets composed exclusively of cash and security positions, with equity as the residual.

These restrictions ensure that the only nontrivial dynamics in the system are those generated by the interaction of cash flows, default events, and the dealer/VBT pricing and inventory mechanisms.

7 Balance–Sheet mechanics by Event

Scope and conventions. All instruments are *tickets* (claims) of face value $S > 0$ (often $S = 1$) that pay at their maturity timepoint. Trades occur in buckets $b \in \{\mathrm{Long}, \mathrm{Mid}, \mathrm{Short}\}$ and settle in the preselected means of payment (cash). Dealers and value-based traders (VBTs) hold cash and tickets only; they have no debt liabilities in this module; equity is residual and omitted from the tables. Quantities of tickets traded are integers $q \in \mathbb{N}$. Prices are per ticket; the cash leg of a trade for q tickets at price p is $q \cdot p$.

Default rule (proportional recovery). At a maturity timepoint t and for a given issuer i, let $N_i(t)$ be the total number of tickets on i that mature at t, so total due is $D_i(t) = S \cdot N_i(t)$. If issuer cash is $C_i(t)$, then:

$$R_i(t) \;=\; \min\!\left(1,\; \frac{C_i(t)}{D_i(t)}\right).$$

Each claimant holding q_h maturing tickets on i receives $q_h \cdot S \cdot R_i(t)$; all those tickets are then deleted and the issuer's post–settlement cash becomes $C_i(t^+) = C_i(t) - R_i(t)\,D_i(t)$ (which is 0 when $R_i(t) < 1$ and $C_i(t) - D_i(t)$ when $R_i(t) = 1$).

Notation used in tables. (I, b) denotes "a ticket on issuer I in bucket b." Dashes (—) indicate no entry. Equity is implicit.

Event 1 — Trader sells a claim to the dealer (dealer buys q at price p)

Starting balance sheets (slice):

Entity	Assets	Liabilities
Trader T	Cash C_T; q tickets (I, b)	—
Dealer^b	Cash C_D; Inventory x tickets in b	—

After trade at p:

Entity	Assets	Liabilities
Trader T	Cash $C_T + q\,p$; 0 of (I, b)	—
Dealer^b	Cash $C_D - q\,p$; Inventory $x + q$ in b	—

Event 2 — Trader buys a claim from the dealer (dealer sells q at price p)

Starting:

Entity	Assets	Liabilities
Trader T	Cash C_T; 0 of (I, b)	—
Dealer^b	Cash C_D; Inventory x $(\geq q)$ in b	—

After trade at p:

Entity	Assets	Liabilities
Trader T	Cash $C_T - q\,p$; q tickets (I, b)	—
Dealer^b	Cash $C_D + q\,p$; Inventory $x - q$ in b	—

Event 3 — Repaying debt to the dealer (issuer pays in full)

Assume at t the dealer holds q maturing tickets on I, and the issuer has $C_I(t) \geq qS$ *and* $C_I(t) \geq D_I(t)$ so $R_I(t) = 1$.

Starting:

Entity	**Assets**	**Liabilities**
Issuer I	Cash $C_I(t)$	q maturing tickets (face S each)
Dealer^b	Cash C_D; q maturing tickets (I, b)	—

After settlement:

Entity	**Assets**	**Liabilities**
Issuer I	Cash $C_I(t) - qS$	Those q liabilities removed
Dealer^b	Cash $C_D + qS$; 0 of those tickets	—

Event 4 — Repaying debt to the value-based trader (issuer pays in full)

Same as Event 3 with owner = VBT^b.

Starting → After:

Entity	**Assets**	**Liabilities**
Issuer I	Cash $C_I(t) \to C_I(t) - qS$	Those q liabilities removed
VBT^b	Cash $C_V \to C_V + qS$; q maturing tickets $(I, b) \to 0$	—

Event 5 — Repaying debt to another trader (issuer pays in full)

Same as Event 3 with owner = some trader K.

Starting → After:

Entity	**Assets**	**Liabilities**
Issuer I	Cash $C_I(t) \to C_I(t) - qS$	Those q liabilities removed
Trader K	Cash $C_K \to C_K + qS$; q maturing tickets $(I, b) \to 0$	—

Event 6 — Defaulting to the dealer (proportional recovery)

At t, issuer I has $C_I(t) < D_I(t)$ so $R := R_I(t) \in (0, 1)$. Let the dealer hold q_D of the $N_I(t)$ maturing tickets.

Starting:

Entity	Assets	Liabilities
Issuer I	Cash $C_I(t)$	All $N_I(t)$ maturing tickets (face S)
Dealer^b	Cash C_D; q_D maturing tickets (I, b)	—

After proportional recovery:

Entity	Assets	Liabilities
Issuer I	Cash 0	All $N_I(t)$ liabilities deleted; default recorded
Dealer^b	Cash $C_D + q_D\,S\,R$; 0 of those tickets	—

Remark. If the dealer is the *only* holder at t ($N_I(t) = q_D$), then $R = C_I(t)/(q_D S)$ and the dealer receives exactly $C_I(t)$.

Event 7 — Defaulting to the value-based trader (proportional recovery)

Same setting as Event 6 with owner share q_V held by VBT^b; the issuer's R is the same for all claimants.

After proportional recovery:

Entity	Assets	Liabilities
Issuer I	Cash 0	All $N_I(t)$ liabilities deleted; default recorded
VBT^b	Cash $C_V + q_V\,S\,R$; those q_V tickets $\to$ 0	—

Event 8 — Defaulting to another trader (proportional recovery)

Same as Event 6 with owner = trader K holding q_K of the $N_I(t)$ maturing tickets.

After proportional recovery:

Entity	Assets	Liabilities
Issuer I	Cash 0	All $N_I(t)$ liabilities deleted; default recorded
Trader K	Cash $C_K + q_K\,S\,R$; those q_K tickets $\to$ 0	—

Event 9 — Dealer lays off *debt* with the VBT (pass-through at outside bid)

Customer sells q tickets (I, b); dealer pins to outside bid B_b and immediately offsets with VBT^b at B_b. Dealer's state is unchanged.

Starting:

Entity	Assets	Liabilities
Trader T	Cash C_T; q tickets (I, b)	—
Dealer^b	Cash C_D; Inventory x in b	—
VBT^b	Cash C_V; Inventory x_V in b	—

After pass-through at B_b:

Entity	Assets	Liabilities
Trader T	Cash $C_T + q\,B_b$; 0 of (I, b)	—
Dealer^b	Cash C_D; Inventory x	—
VBT^b	Cash $C_V - q\,B_b$; Inventory x_V+q in b	—

Event 10 — Dealer lays off *money* with the VBT (pass-through at outside ask)

Customer buys q tickets; dealer pins to outside ask A_b and immediately offsets with VBT^b at A_b. Dealer's state is unchanged.

Starting:

Entity	Assets	Liabilities
Trader T	Cash C_T; 0 of (I, b)	—
Dealer^b	Cash C_D; Inventory x in b	—
VBT^b	Cash C_V; Inventory x_V $(\geq q)$ in b	—

After pass-through at A_b:

Entity	Assets	Liabilities
Trader T	Cash $C_T - q\,A_b$; q tickets (I, b)	—
Dealer^b	Cash C_D; Inventory x	—
VBT^b	Cash $C_V + q\,A_b$; Inventory x_V-q in b	—

Event 11 — As time passes, Long-bucket Dealer sells to Mid-bucket Dealer

When a held ticket crosses the Long/Mid boundary, represent rebucketing as an internal sale of q tickets at the Mid-bucket mid price M_{Mid} from Dealer$^{\text{Long}}$ to Dealer$^{\text{Mid}}$ (double-entry preserving).

Starting:

Entity	Assets	Liabilities
Dealer$^{\text{Long}}$	Cash C_L; q tickets (I, Long)	—
Dealer$^{\text{Mid}}$	Cash C_M; Inventory x_M in Mid	—

After internal sale at M_{Mid}*:*

Entity	Assets	Liabilities
Dealer$^{\text{Long}}$	Cash $C_L + q\,M_{\text{Mid}}$; those q tickets $\to 0$	—
Dealer$^{\text{Mid}}$	Cash $C_M - q\,M_{\text{Mid}}$; Inventory $x_M + q$ in Mid	—

Event 12 — As time passes, Long-bucket VBT sells to Mid-bucket VBT

Analogous to Event 11 for VBT buckets.

Starting $\to$ *After (at* M_{Mid}*):*

Entity	Assets	Liabilities
VBT$^{\text{Long}}$	Cash $C_L \to C_L + q\,M_{\text{Mid}}$; those q tickets $\to 0$	—
VBT$^{\text{Mid}}$	Cash $C_M \to C_M - q\,M_{\text{Mid}}$; Inventory $x_M \to x_M + q$ in Mid	—

Consistency checks. (i) Every trade shows mirrored cash/ticket legs. (ii) Settlement uses the proportional recovery rule; defaults extinguish all maturing tickets and set issuer cash to zero when $R < 1$. (iii) Pass-through (Events 9–10) leaves the dealer's state unchanged. (iv) Internal rebucketing via mid-price transfers (Events 11–12) preserves double-entry and keeps equity measurement consistent across buckets.

8 Dealer Kernel (Per Bucket)

Fix a bucket b and suppress the superscript b for clarity; everything in this section is per bucket.

8.1 Inputs

Per bucket we have:

- Standard ticket size $S > 0$ (same across buckets);
- VBT mid M and spread O, with outside quotes $A = M + \frac{O}{2}$, $B = M - \frac{O}{2}$;
- Dealer balances: securities a (number of tickets) and cash C.

8.2 Capacity and ladder

Mid–valued inventory:

$$V = Ma + C.$$

Maximum number of buy tickets fundable without borrowing:

$$K^* = \left\lfloor \frac{V}{M} \right\rfloor.$$

One–sided capacity in units of security (face):

$$X^* = SK^*.$$

Define the one–sided ladder:

$$x \in \{0, S, 2S, \ldots, X^*\},$$

with $N = X^*/S + 1$ rungs.

Guard. If $M \leq M_{\min}$ (small positive threshold), set $X^* := 0$ and pin quotes to the outside. Otherwise compute $K^* = \lfloor V/M \rfloor$ as above.

8.3 Layoff probability and inside width

Given a symmetric reflecting random walk on the ladder, the layoff (outside execution) probability per ticket is

$$\lambda = \frac{1}{N} = \frac{S}{X^* + S}.$$

Competition pins the inside width:

$$I = \lambda O = \frac{S}{X^* + S}\, O.$$

8.4 Inventory–sensitive midline

The mean inside midline $p(x)$ is linear in inventory and anchored one step beyond the edges via "ghost" points. The closed form is:

$$p(x) = M - \frac{O}{X^* + 2S}\left(x - \frac{X^*}{2}\right).$$

8.5 Interior quotes and clipping

Interior (unclipped) quotes:

$$a(x) = p(x) + \frac{I}{2}, \qquad b(x) = p(x) - \frac{I}{2}.$$

Clipped to outside:

$$a_c(x) = \min\{A, a(x)\}, \qquad b_c(x) = \max\{B, b(x)\}.$$

8.6 Commit–to–quote feasibility and pins

We impose no shorting and no borrowing. We must be able to honor one ticket at the posted quote.

Customer BUY (dealer sells).

- Feasible as an interior sale if $x \geq S$.
- Otherwise, pin the ask to outside A: use price $p = A$. Under atomic clearing, the dealer immediately offsets with the VBT at A; the dealer's state does not change.

Customer SELL (dealer buys).
- Interior feasible if $x + S \le X^*$ and $C \ge b_c(x)$.
- Boundary pass–through at outside bid B is always feasible under atomic clearing (dealer executes customer at B and simultaneously offsets with VBT at B); the dealer's state does not change: $x' = x,\ C' = C$.

8.7 State updates for a single ticket

Let p be the price at which the customer leg executes (either an inside quote or an outside pin). The baseline updates for one ticket are:

Customer SELL to dealer (dealer buys).
- Interior execution at price $p \in \{b_c(x)\}$:

$$x' = x + S, \qquad C' = C - p.$$

- Boundary pass–through at outside bid B: execute customer at B, immediately lay off to VBT at B; net effect on dealer state:

$$x' = x, \qquad C' = C.$$

Customer BUY from dealer (dealer sells).
- Interior execution at price $p \in \{a_c(x)\}$:

$$x' = x - S, \qquad C' = C + p.$$

- Boundary pass–through at outside ask A: execute customer at A, immediately offset with VBT at A; net effect:

$$x' = x, \qquad C' = C.$$

9 Initialization of Dealer State

Per bucket:
1. Choose target capacity X^*_{target} and outside parameters M, O.
2. Compute implied λ and I:

$$\lambda = \frac{S}{X^*_{\text{target}} + S}, \qquad I = \lambda O.$$

3. Set initial inventory to the middle of the shelf:

$$x_0 = \frac{X^*_{\text{target}}}{2}.$$

4. Set initial securities $a_0 = x_0/S$, initial cash

$$C_0 = Ma_0 = M\frac{x_0}{S}.$$

   This yields equal mid–valued amounts in cash and securities:

$$Ma_0 = C_0 = M\frac{X^*_{\text{target}}}{2S}.$$

5. Recompute X^* from (a_0, C_0, M, S) using the capacity formula. This yields $X^* = X^*_{\text{target}}$, so the ladder is consistent.

Initial cross–sectional allocation (per bucket). Let $N^b_{\rm tot}$ be the number of bucket–b tickets at $t = 0$. Reassign ownership so that Dealer^b holds $\lfloor 0.25\, N^b_{\rm tot} \rfloor$, VBT^b holds $\lfloor 0.50\, N^b_{\rm tot} \rfloor$, and the remaining tickets stay with the original traders (random, or proportional to their initial holdings). Book at price M^b_0 so total cash and equity are internally consistent.

10 Updating VBT Anchors from Bucket Defaults

At the end of each period t, after settlement but before dealer pre–computation at $t+1$, we update the value-based trader (VBT) anchors for each bucket $b \in \{\text{Long}, \text{Mid}, \text{Short}\}$: the VBT mid M^b_{t+1} and the VBT spread O^b_{t+1}. The update uses only information from period t: the current anchors (M^b_t, O^b_t) and the realized losses on tickets that were in bucket b and matured during period t.

Bucket-level loss rate. Fix a bucket b and period t. Let $\mathcal{T}^{b,\text{mat}}_t$ be the set of tickets that:

- belonged to bucket b at the *start* of period t, and
- had `maturity_time` $= t$ (so they matured and were settled in t).

Each such ticket $k \in \mathcal{T}^{b,\text{mat}}_t$ promises face S and is associated to an issuer $i(k)$ with recovery rate $R_{i(k)}(t)$ determined by the proportional recovery rule in the settlement step:

$$R_{i(k)}(t) = \min\left(1,\ \frac{C_{i(k)}(t)}{D_{i(k)}(t)}\right).$$

The realized payoff on ticket k is $S\,R_{i(k)}(t)$, so the loss on k is $S\big(1 - R_{i(k)}(t)\big)$. We define the *bucket loss rate* for (b, t) as

$$\ell^b_t \;:=\; \begin{cases} 0, & \text{if } \mathcal{T}^{b,\text{mat}}_t = \emptyset, \\[2ex] \dfrac{\displaystyle\sum_{k \in \mathcal{T}^{b,\text{mat}}_t} S\big(1 - R_{i(k)}(t)\big)}{\displaystyle\sum_{k \in \mathcal{T}^{b,\text{mat}}_t} S}, & \text{otherwise.} \end{cases} \tag{1}$$

By construction $\ell^b_t \in [0, 1]$. A value $\ell^b_t = 0$ means no loss (default-free repayment of all tickets from bucket b in t), while $\ell^b_t = 1$ means full loss on all such tickets.

Linear update rule for mid and spread. Given the current anchors (M^b_t, O^b_t) and the bucket loss rate ℓ^b_t, we define the next-period VBT mid M^b_{t+1} and spread O^b_{t+1} by

$$M^b_{t+1} = M^b_t \;-\; \phi^b_M\, \ell^b_t, \tag{2}$$

$$O^b_{t+1} = O^b_t \;+\; \phi^b_O\, \ell^b_t, \tag{3}$$

where $\phi^b_M > 0$ and $\phi^b_O > 0$ are bucket-specific sensitivity parameters.

- The term $-\phi^b_M\, \ell^b_t$ makes the VBT mid *fall* when there are losses (higher perceived credit risk $\Rightarrow$ lower price).
- The term $+\phi^b_O\, \ell^b_t$ makes the VBT spread *widen* when there are losses (higher perceived risk $\Rightarrow$ wider bid–ask).
- If no bucket losses occur in t ($\ell^b_t = 0$), the VBT anchors remain unchanged: $M^b_{t+1} = M^b_t$, $O^b_{t+1} = O^b_t$.

To enforce non-negative bids and a minimum spread, we can optionally clip:

$$O^b_{t+1} \leftarrow \max\{O^b_{\min},\, O^b_{t+1}\}, \tag{4}$$

$$B^b_{t+1} := M^b_{t+1} - \tfrac{1}{2}O^b_{t+1}, \qquad A^b_{t+1} := M^b_{t+1} + \tfrac{1}{2}O^b_{t+1}, \tag{5}$$

$$B^b_{t+1} \leftarrow \max\{0,\, B^b_{t+1}\}, \tag{6}$$

and then set the outside quotes by

$$A^b_{t+1} = M^b_{t+1} + \frac{O^b_{t+1}}{2}, \qquad B^b_{t+1} = M^b_{t+1} - \frac{O^b_{t+1}}{2}. \tag{7}$$

Position in the event loop. This update is applied at the *end* of period t to produce next–period anchors:

1. After settlement in t, compute ℓ^b_t for each bucket b via (1).
2. Update (M^b_{t+1}, O^b_{t+1}) via (2)–(3) and form (A^b_{t+1}, B^b_{t+1}) via (7).
3. Use (M^b_{t+1}, O^b_{t+1}) in dealer pre–computation at time $t+1$.

This yields a simple, purely backward-looking rule: VBT mids and spreads for each bucket move only when there were losses in that bucket in the current period; otherwise they remain fixed.

11 Agent Trading Policies

These are simple heuristics that determine when a ring agent trades with the dealer.

11.1 Liquidity policy (selling tickets)

For each agent i at period t:

1. Compute net cash position after funding all obligations due at t:

$$\text{shortfall}_i = \max\{0, \text{payments\_due}_i(t) - \texttt{cash\_i}(t)\}.$$

2. While $\text{shortfall}_i > 0$ and agent i owns tickets:
   (a) Select a ticket to sell:
   - priority rule: sell ticket with the shortest remaining time–to–maturity (closest maturity first);
   - if multiple tickets in different buckets satisfy this, choose the one whose bucket dealer currently offers the highest bid.

   (b) Query the appropriate bucket dealer for current bid $b_c(x)$.
   (c) Attempt to execute a SELL of one ticket:
   - if interior *dealer BUY* is feasible or pinned BUY at outside B is feasible, execute at quoted price p;
   - update dealer state (x, C), agent cash, and ticket owner.

   (d) Reduce $\texttt{shortfall}_i by p$.

11.2 Investment policy (buying tickets)

For each agent i at period t:

1. Compute earliest liability date $T^{\min}_i > t$. If no future obligations, treat $T^{\min}_i = +\infty$.
2. If

$$T^{\min}_i - t \geq H \quad \text{and} \quad \texttt{cash\_i}(t) > \text{buffer\_B},$$

   the agent is willing to invest one ticket this period.

3. Choose the preferred bucket in order:
   - Short $\to$ Medium $\to$ Long,
   - skip buckets whose dealer is currently at outside pins on the relevant side (optional refinement).
4. Query the chosen bucket dealer for ask $a_c(x)$ and attempt to BUY one ticket: execute only if $\text{cash}_i(t) \geq p$ (with $p = a_c(x)$ or the outside A if pinned) and either (i) interior dealer SELL is feasible ($x \geq S$), or (ii) a pinned SELL at outside A is available (pass–through). Execute at price p, deduct p from cash_i, and *transfer one existing dealer ticket* (issuer + maturity) to agent i.

Ticket creation vs. transfer for BUYs. There are two options for implementation:
- Track the dealer as actual owner of specific tickets and transfer one existing dealer ticket to the buyer; This requires individual tickets on the dealer book.

Requirement. For settlement consistency, dealers and VBTs MUST track inventory at the ticket level (issuer + maturity). Implement the BUY leg by transferring one actual dealer ticket to the buyer. The "materialize generic ticket" option is disallowed in the baseline.

Issuer selection for BUYs. If trader i holds claims on issuer $k(i)$, any BUY by i must be of issuer $k(i)$. If i holds none, draw an issuer from the dealer's current bucket inventory (proportional to shares outstanding) and set $k(i)$.

12 Simulation Event Loop (Per Period)

For each time step t:
1. **Update maturities and buckets**
   - For each ticket, decrement remaining τ and recompute bucket membership (Short/Medium/Long) according to fixed ranges.
2. **Dealer pre–computation (per bucket)**
   (a) Given $(M_t^b, O_t^b, S, a_t^b, C_t^b)$, compute capacity $X_t^{*,b}$;
   (b) compute $\lambda_t^b = S/(X_t^{*,b} + S)$ and $I_t^b = \lambda_t^b O_t^b$;
   (c) for current inventory x_t^b, compute midline $p^b(x_t^b)$;
   (d) compute unclipped and clipped quotes $a_c^b(x_t^b), b_c^b(x_t^b)$.
3. **Compute trading eligibility sets**
   - For each agent i, compute:
     - liquidity shortfall
     $$\text{shortfall}_i(t) = \max\{0,\ \text{payments\_due}_i(t) - \text{cash}_i(t)\};$$
     - earliest future liability date $T_i^{\min}(t) > t$ (or $+\infty$ if none).
   - An agent is *eligible to sell* one ticket at t if $\text{shortfall}_i(t) > 0$ and i owns at least one ticket. Collect these in a set $\mathcal{S}_t$.
   - An agent is *eligible to buy* one ticket at t if
     $$T_i^{\min}(t) - t \ \geq\ H \quad \text{and} \quad \text{cash}_i(t) > \text{buffer}_B.$$
     Collect these in a set $\mathcal{B}_t$.
   - Within period t, each agent can execute at most one liquidity–motivated SELL and at most one investment–motivated BUY, so once an actual trade occurs for i on a given side, remove i from the corresponding set $\mathcal{S}_t$ or $\mathcal{B}_t$.
4. **Randomized one–ticket order flow**
   - Fix a parameter $\pi_{\text{sell}} \in (0, 1)$ (probability that the next arrival is a SELL) and an integer $N_{\max} \geq 1$ (maximum arrivals per period).
   - Initialize arrival counter $n = 1$.

- While $n \le N_{\max}$ and $(\mathcal{S}_t \cup \mathcal{B}_t) \neq \emptyset$:
  (a) Draw a Bernoulli variable $Z_n \sim \text{Bernoulli}(\pi_{\text{sell}})$.
  (b) *If* $Z_n = 1$ (SELL side preferred):
    - If $\mathcal{S}_t \neq \emptyset$:
      * pick an agent $i \in \mathcal{S}_t$ uniformly at random;
      * let i choose which ticket to sell (shortest remaining maturity first; if several buckets tie, choose the one whose bucket dealer has the highest bid);
      * query the relevant bucket dealer for $b_c^b(x_t^b)$ and check feasibility (capacity, pass–through pins, etc.);
      * if feasible, execute a *customer SELL*: i sells one ticket to the dealer at the appropriate price; update ticket ownership, dealer inventory and cash, and the agent's cash and shortfall;
      * remove i from $\mathcal{S}_t$.
    - else if $\mathcal{S}_t = \emptyset$ but $\mathcal{B}_t \neq \emptyset$, treat this arrival as a BUY instead (fallback).
  (c) *If* $Z_n = 0$ (BUY side preferred):
    - If $\mathcal{B}_t \neq \emptyset$:
      * pick an agent $i \in \mathcal{B}_t$ uniformly at random;
      * let i choose a bucket to buy from (e.g. Short $\to$ Mid $\to$ Long, optionally skipping buckets where the dealer is pinned at the outside ask);
      * query the chosen bucket dealer for $a_c^b(x_t^b)$ and check feasibility (dealer must have inventory or pass–through to VBT is activated);
      * if feasible, execute a *customer BUY*: dealer sells one ticket to i at the appropriate price; update ticket ownership and cash;
      * remove i from $\mathcal{B}_t$.
    - else if $\mathcal{B}_t = \emptyset$ but $\mathcal{S}_t \neq \emptyset$, treat this arrival as a SELL instead (fallback).
  (d) After any executed trade, update all affected dealer states and, if needed, re-compute quotes $a_c^b(x_t^b), b_c^b(x_t^b)$ for the relevant bucket(s).
  (e) Increment $n \leftarrow n + 1$.
  (f) If no feasible execution, the agent remains in the eligibility set

5. **Settlement with proportional recovery**
   - For each issuer i, collect all tickets with `maturity_time` $= t$. Let $N_i(t)$ be their number and $D_i(t) = S \cdot N_i(t)$ the total due.
   - Let issuer cash be $C_i(t)$. Define the recovery rate

$$R_i(t) \;=\; \min\!\left(1,\; \frac{C_i(t)}{D_i(t)}\right).$$

   - For each holder h with q_h maturing tickets on issuer i:
     - pay $q_h \cdot S \cdot R_i(t)$ from issuer i to holder h;
     - delete those q_h tickets from holder h's assets and from issuer i's liabilities.
   - Update issuer cash to

$$C_i(t^+) \;=\; C_i(t) - R_i(t) D_i(t),$$

     so that $C_i(t^+) = 0$ when $R_i(t) < 1$ (default with partial recovery) and $C_i(t^+) = C_i(t) - D_i(t)$ when $R_i(t) = 1$ (full repayment).

6. **Update VBT anchors (optional)**
   - For each bucket b, compute the bucket loss rate ℓ_t^b from period t as in Section 10, and set (M_{t+1}^b, O_{t+1}^b) using the linear update rule (2)–(3) (or keep $(M_{t+1}^b, O_{t+1}^b) = (M_t^b, O_t^b)$ if anchors are held fixed).

13 Implementation Notes and Interfaces

13.1 Recommended module structure

At minimum, implement:

`DealerBucket` (per bucket).

- State: $\{S, M, O, A, B, a, C, x, X^*, \lambda, I\}$.
- Methods:
  - `recompute_capacity_and_quotes()`: updates $X^*, \lambda, I, p(x), a_c(x), b_c(x)$ from current state;
  - `quote_bid()` / `quote_ask()`: return current $b_c(x)$ or $a_c(x)$;
  - `can_buy_one()` / `can_sell_one()`: feasibility checks for BUY/SELL of one ticket given (x, C) and pass–through rules;
  - `execute_customer_sell()`: update (x, C) for a customer SELL (dealer buys one ticket);
  - `execute_customer_buy()`: update (x, C) for a customer BUY (dealer sells one ticket).

`Agent`.

- State: cash, list of tickets owned, list of obligations (issuer, face, maturity), etc.
- Methods:
  - `compute_shortfall(t)`: returns $\text{shortfall}_i(t)$;
  - `compute_earliest_liability(t)`: returns $T_i^{\min}(t)$;
  - `eligible_to_sell(t)` / `eligible_to_buy(t)`: implement the liquidity and investment conditions;
  - `choose_ticket_to_sell()`: select which ticket to sell (shortest maturity first, etc.);
  - `choose_bucket_to_buy()`: select a bucket according to the preferred order and any skipping rules.

`Ticket`.

- Fields: issuer, owner, face = S, maturity_time, remaining_tau, bucket_id.

13.2 Parameters for random order flow

Introduce global (or configurable) parameters:

- $\pi_{\text{sell}} \in (0, 1)$: probability that a given arrival is a SELL (liquidity–motivated customer selling to the dealer);
- $N_{\max} \in \mathbb{N}$: maximum number of one–ticket arrivals processed per period.

13.3 Testing

Before integrating into the full Kalecki ring, test each `DealerBucket` and the order–flow module in isolation:

- Verify that capacity X^* computed from (M, S, a, C) matches design.
- Check that $\lambda = S/(X^* + S)$ and $I = \lambda O$.
- Confirm that midline $p(x)$ is linear in x and anchored correctly by outside quotes.
- Check feasibility and pins near the edges (quotes flatten at outside levels one ticket before running out of cash or securities).
- Check that pass–through trades leave the dealer's (x, C) unchanged.
- With fixed π_{sell}, simulate many periods and verify that the empirical fraction of SELL vs. BUY arrivals converges to π_{sell} (subject to eligibility constraints).

- In settlement, verify that full repayment and default with partial recovery behave as specified: cash paid to all claimants is $R_i(t)D_i(t)$ and issuer cash updates to $C_i(t^+) = C_i(t) - R_i(t)D_i(t)$.