Case Study: Intelligent Systems in Production

RL-Driven Warehouse Item Relocation Optimization

Prof. Hamidreza Heidari

Binil Sajeev, Abdelraheem Zekry, Orlando Manrique, Arthur Reis

0 Table of Contents

1 Problem Statement

2 Modeling and Simulation Workflow

3 Results and Validation - RL Final Results

3 Results and Validation - Geometric Validation

3 Results and Validation - Metrics and Validation

4 Discussion and Analysis - RL Results

4 Discussion and Analysis - Heuristic Results

4 Discussion and Analysis - Conclusion

5 References

1 Problem Statement

- Project Title: **RL-Driven Warehouse Item Relocation Optimization**
- Team:
  - Binil Sajeev, Abdelraheem Zekry Reinforcement Learning & Optimization
  - Orlando Manrique Geometric Modeling & Simulation
  - Arthur Reis Data Engineering, Analytics & Validation
- Industrial Problem: **Limited space and dynamic inventory make manual warehouse allocation impossibly complex, leading to congestion and inefficiency.**
- Goal: **Develop an RL-based system that finds an optimal trade-off between high storage density and fast warehouse operations.**

2 Modeling and Simulation Workflow

Workflow and Key Steps:

Problem Definition → Synthetic Data Generation → Preprocessing & Segmentation → Constraint & Metric Libraries →

Synthetic Data Generation: Parts (dims, demand, boxes) + Locations (bin dims, coords)

Baseline Allocation → Optimization Engines (2 Methods) → Validation & Rule Enforcement → Results & Comparison

Baseline Allocation: Random SKU & bin order → place into feasible bin (messy layout)

Optimization Engines (2 Methods): From baseline state, generate two improved layouts:

(A) Heuristic relocation

(B) RL relocation

Results & Comparison: Baseline vs Heuristic vs RL → charts + conclusion (best method)

2 Modeling and Simulation Workflow

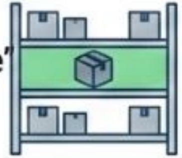
Initial State & Metrics

```
--- Initial State ---

================ PROCESSING: Initial State ==============
Weight Violations          : 0 (OK)
Misplaced A-Items          : 17 (!)

--- Scoring Breakdown (Avg per Occupied Bin) ---
Avg Combined Score         : 261.0
  > Zone Reward            : 27.6
  > Util Reward            : 285.2
  > Dist Penalty           : -51.8

Avg Util (Occupied)        : 35.65%
%Occupied Bins             : 48.7%  (174/357)
```

Zone Reward: Binary Bonus System

Fast Zone: (+500 points)
Class A item in front 25%
($X \leq 0.25X$(max)).

Ergo Zone: (+500 points)
Heavy item (>15kg) in "Golden Zone"
(700mm - 1500mm).



Key Metrics & Formulas

Avg Combined Score:

$$S_{i,b} = \text{ZoneReward} + \text{UtilReward} - \text{DistPenalty}$$

Utilization Reward: A score from 0–1000 representing bin fullness.

$$\text{UtilReward} = \left(\frac{\text{Volume}_{\text{item}} \times \text{Qty}}{\text{Volume}_{\text{bin}}}\right)$$

Distance Penalty: A cost function based on Manhattan Distance from entrance.

$$\text{DistPenalty} = 0.1 \times (|X_b| + |Y_b - Y_{\text{ent}}|)$$

Avg Util (Occupied): Measures how full bins are (ignoring empty ones).

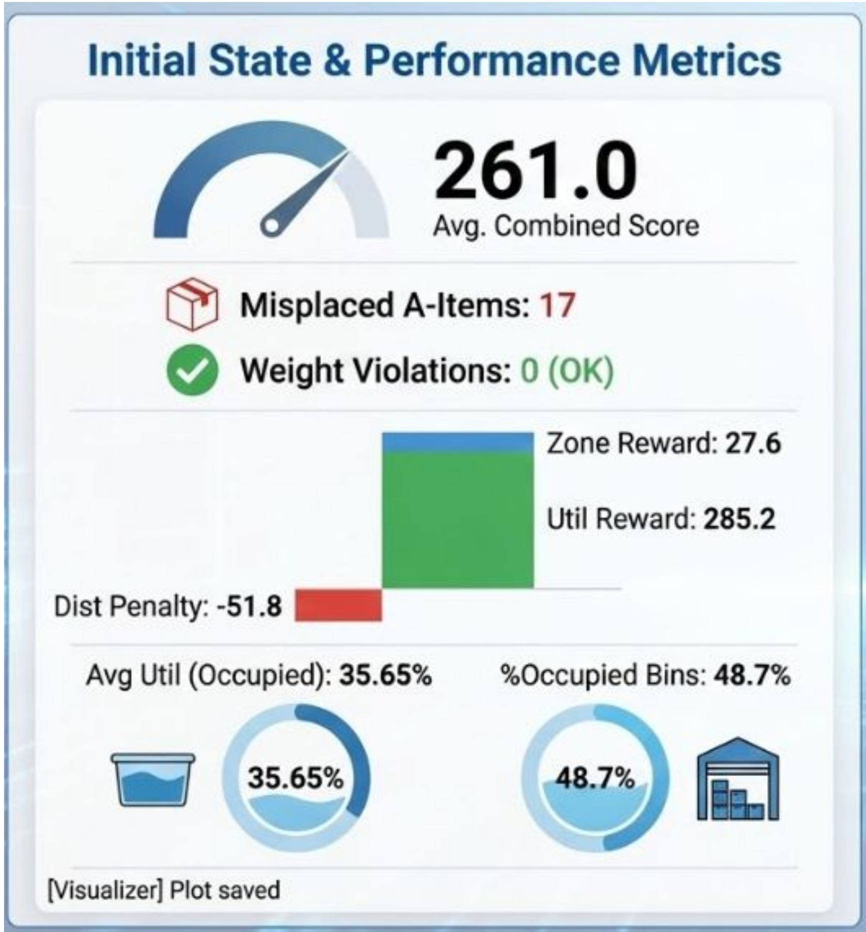
$$\text{Avg Util} = \frac{\sum(\text{Utilization \% of bin } i)}{\text{Total Number of Occupied Bins}}$$

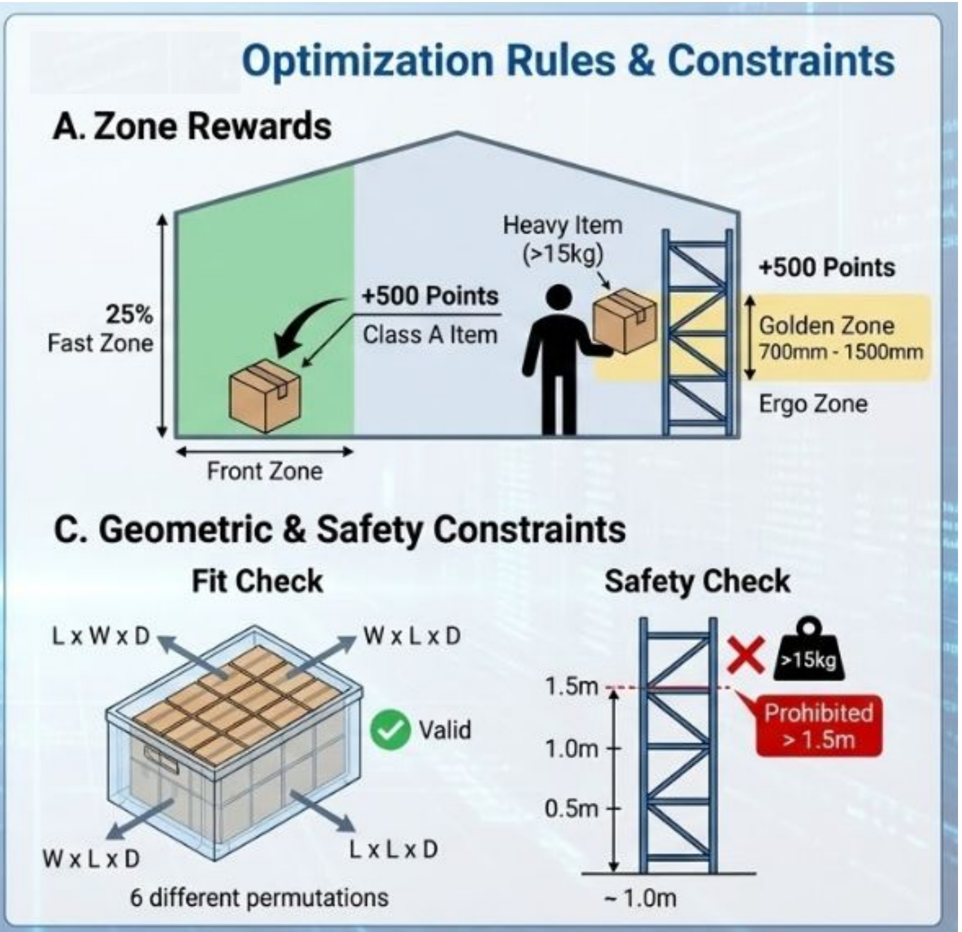
% Occupied Bins: Measures warehouse saturation/fullness.

$(174/357) = 48.7\%$

$$\%\,\text{Occupied} = \frac{\text{Number of Bins containing at least 1 item}}{\text{Total Number of Bins in Warehouse}} \times 100$$

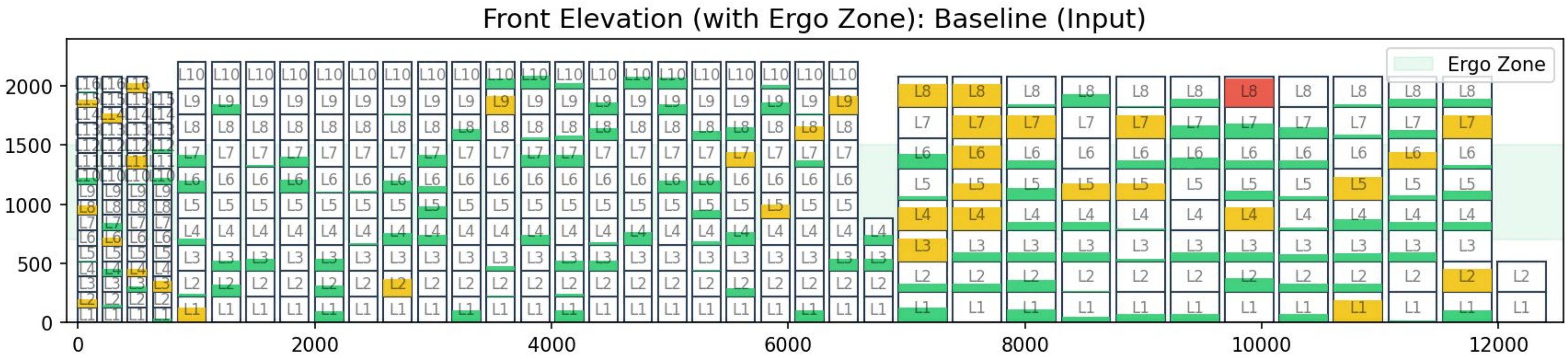
2 Modeling and Simulation Workflow

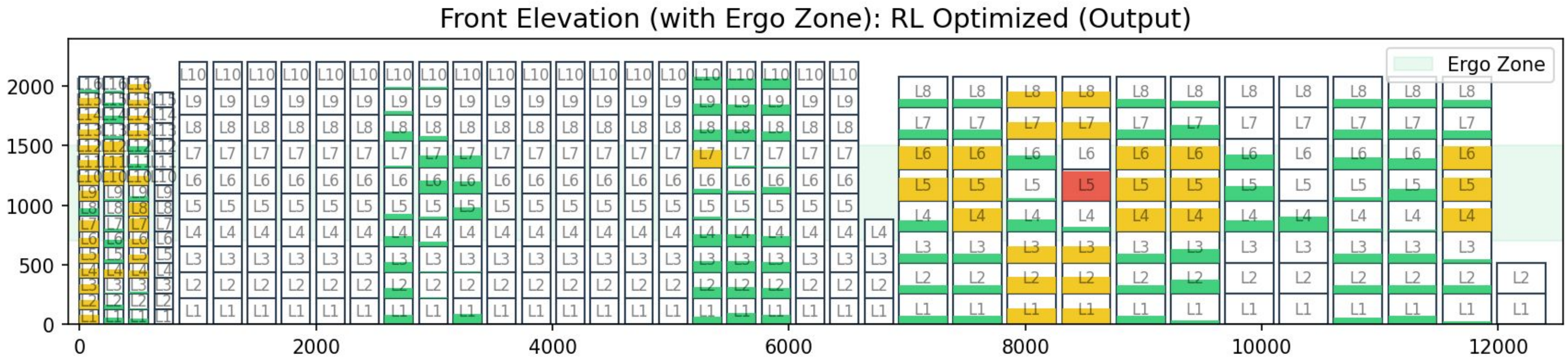




3 Results and Validation - RL Final Results

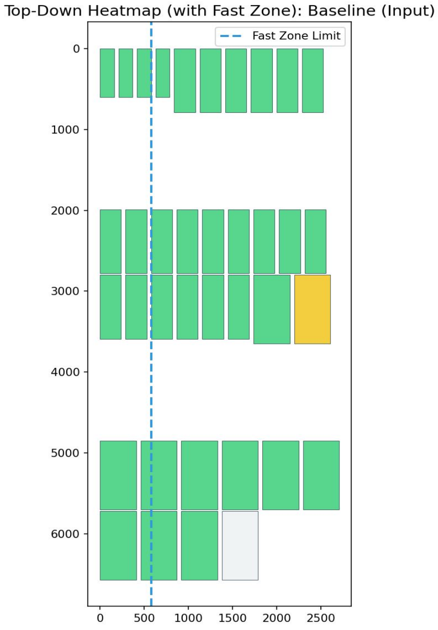
Front Elevation View (Baseline → RL Optimized)

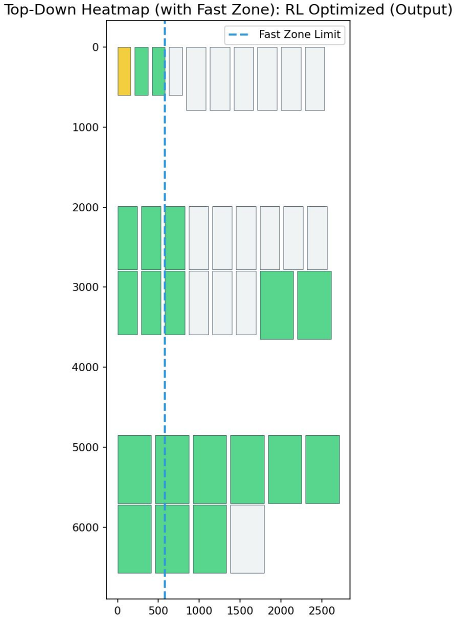




3 Results and Validation - RL Final Results

Top-Down Heatmap (Baseline → RL Optimized)



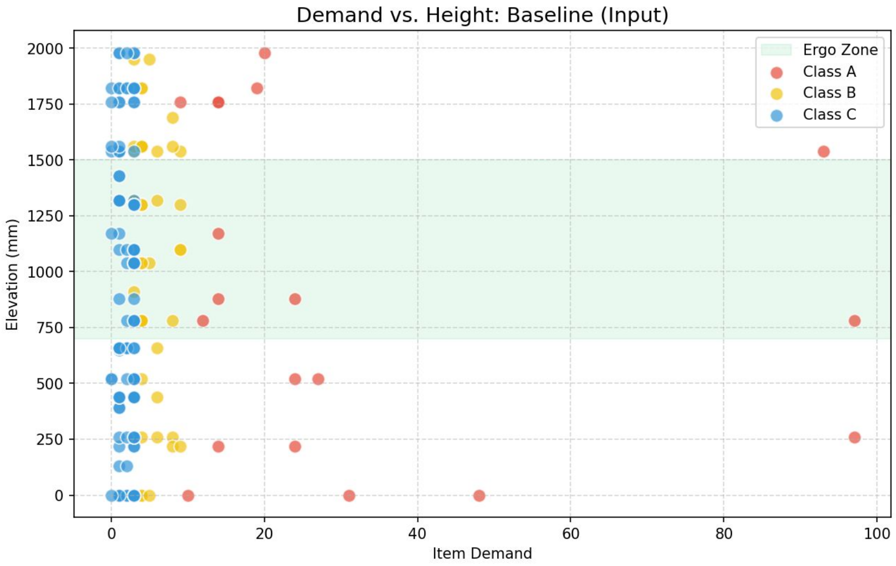


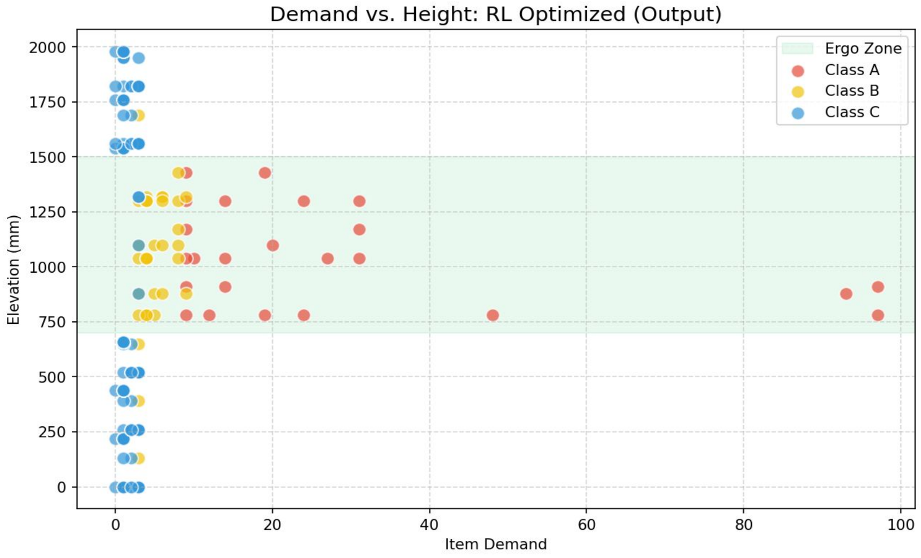
- High-demand items are scattered across both fast and non-fast zones.
- Several premium fast-zone locations are occupied by low-priority or inefficient placements.
- Limited alignment between spatial proximity to dispatch and item demand.
- Indicates rule-based or static allocation without congestion awareness.

- High-demand items are preferentially concentrated within the fast zone.
- Low-demand items are pushed toward non-fast regions, freeing prime locations.
- Clear spatial reorganization without increasing occupied bin count.
- Demonstrates learned prioritization rather than hard-coded zoning.

3 Results and Validation - RL Final Results

Demand vs. Height Scatter (Baseline → RL Optimized)



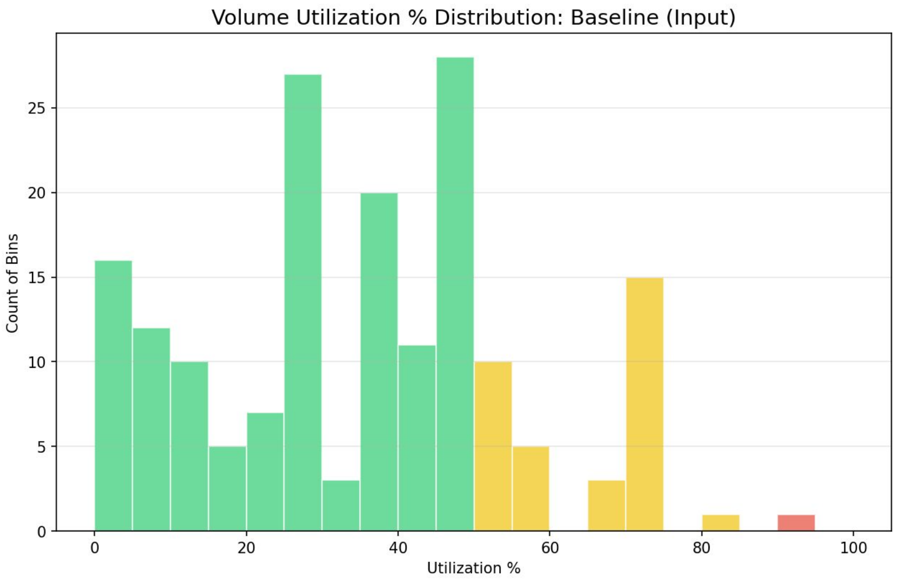


- Weak correlation between item demand and storage height.
- High-demand items appear at both extreme low and high elevations.
- Increased operational effort due to inefficient vertical placement.

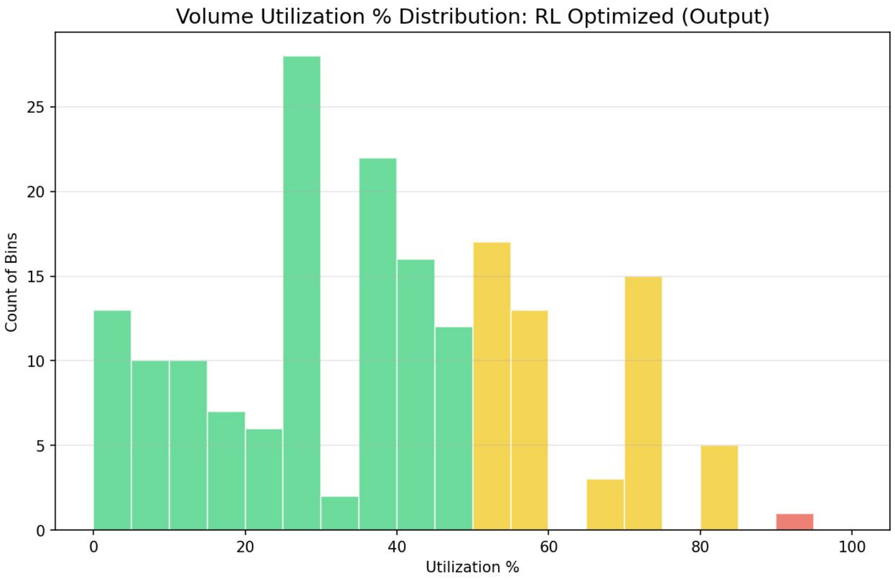
- **Strong clustering of high-demand items within ergonomic height band.**
- **Clear separation between fast-moving and slow-moving SKUs.**
- **Reduced vertical travel for frequent picks.**
- **Indicates congestion-aware and demand-sensitive learning behavior.**

3 Results and Validation - RL Final Results

Volume Utilization Distribution (Baseline → RL Optimized)



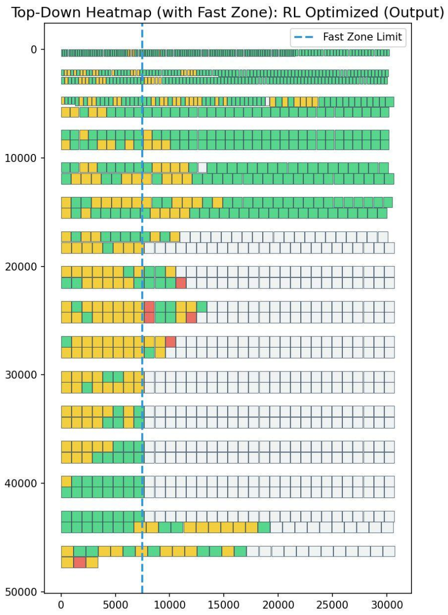
- Wide spread of utilization percentages across bins.
- Many bins are underutilized despite available consolidation opportunities.
- Presence of inefficient partial fills.



- **Utilization distribution shifts toward higher and more consistent values.**
- **Improved consolidation of items within bins.**
- **Increase in average utilization without sacrificing feasibility.**
- **Demonstrates balance between space efficiency and operational constraints.**

3 Results and Validation - RL Final Results

Upscaling Validation: RL Performance on Large-Scale Warehouse (≈6,300 locations)




```
---------- Baseline (Input) ----------
Weight Violations          : 7 (!)
Misplaced A-Items          : 600 (!)

--- Scoring Breakdown (Avg per Occupied Bin) ---
Avg Combined Score         : 214.6
  > Zone Reward            : 30.5
  > Util Reward            : 266.4
  > Dist Penalty           : -61.0

Avg Util (Occupied)        : 33.29%
%Occupied Bins             : 54.2%   (3413/6300)
----------------------------------------------
```

```
---------- RL Optimized (Output) ----------
Weight Violations          : 0 (OK)
Misplaced A-Items          : 146 (!)

--- Scoring Breakdown (Avg per Occupied Bin) ---
Avg Combined Score         : 449.1
  > Zone Reward            : 160.0
  > Util Reward            : 346.2
  > Dist Penalty           : -57.1

Avg Util (Occupied)        : 43.27%
%Occupied Bins             : 63.7%   (4012/6300)
----------------------------------------------
```

Upscaling confirms that the RL agent generalizes beyond prototype layouts, maintaining safety guarantees while improving spatial efficiency under increased complexity.

3 Results and Validation - Geometric Engine

Dimensional Feasibility & Orientations

1. SKUs and storage locations are modeled as rigid 3D cuboids with dimensions in millimeters.
2. SKUs and storage locations are modeled as rigid 3D cuboids with dimensions in millimeters.
3. For each orientation, the maximum number of units that fit along each axis is computed using integer division:

$$n_x = \lfloor L_x / s_x \rfloor$$
$$n_y = \lfloor L_y / s_y \rfloor$$
$$n_z = \lfloor L_z / s_z \rfloor$$

Dimensional Feasibility & Orientations

4. This defines a 3D allocation grid (Nx, Ny, Nz).
5. It calculates each grid capacity and the orientation yielding the largest feasible unit count is selected.
6. If no orientation produces a positive capacity, the SKU is classified as physically infeasible for that location.

3 Results and Validation - Allocation Engine

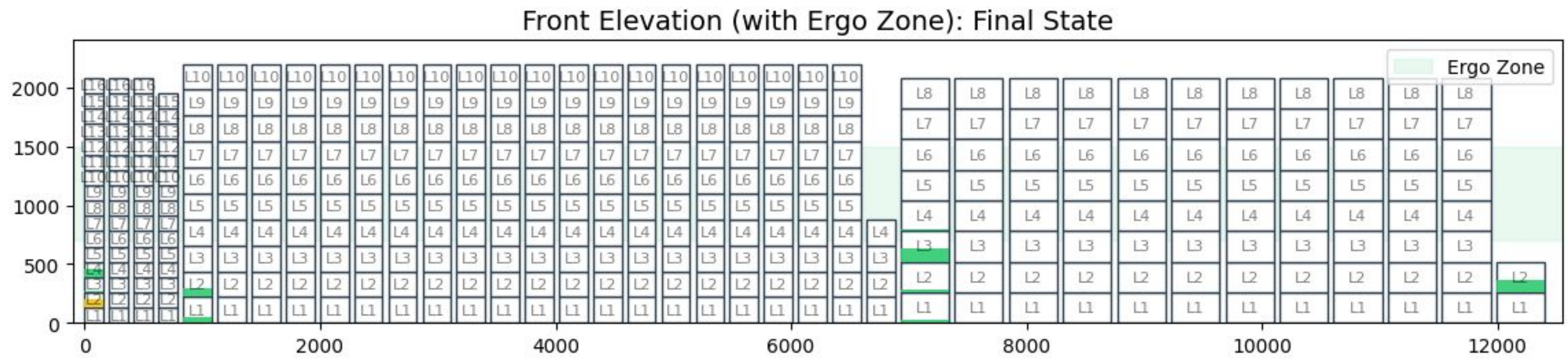
Layered Packing (Grid)

1. Allocation Grid is decomposed into XY layers, stacked along Z axis (height)..
2. The engine can generate 2D occupancy matrices per layer for debugging and validation.
3. Guarantees volume consistency between allocated stock and occupied space
4. All layers must be full, except the last one (can be filled partly).

```
10. Location: A1-00044 | Type: A1 | SKU: 26613318
    Slot position (mm): X=340, Y=0, Z=1430
    Initial allocation: 4 / 10 units
    Grid capacity (X x Y x Z): 2 x 5 x 1
    Allocation layout: 2 x 5 x 0 + 4 units on last layer
    Product orientation (mm): (X=40, Y=126, Z=121)
    Partial layer (Z = 1) in ASCII:
        [X][X]
        [X][X]
        [ ][ ]
        [ ][ ]
        [ ][ ]

11. Location: A1-00028 | Type: A1 | SKU: 23947016
    Slot position (mm): X=170, Y=0, Z=1430
    Initial allocation: 9 / 9 units
    Grid capacity (X x Y x Z): 1 x 9 x 1
    Allocation layout: 1 x 9 x 1
    Product orientation (mm): (X=68, Y=68, Z=68)
    All layers full.
```

3 Results and Validation - Allocation Engine

1. At simulation time t=0, inventory is allocated randomly to avoid structural bias. It is constrained by geometric feasibility.
2. SKUs are processed in descending order of geometric difficulty (largest dimension first).
3. Each SKU is assigned to the first empty location where a valid grid exists.
4. Locations are filled up to maximum feasible capacity or until stock is exhausted.
5. Unallocated SKUs are classified as: No feasible fit anywhere, No free locations, No feasible fit in remaining locations.



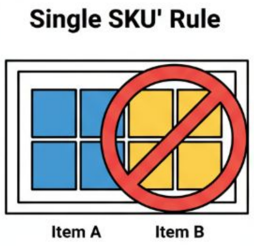
3 Results and Validation - Data Validation

A. Physical Feasibility

- **Geometric Stack Fit:** Verifies Grid(XYZ) × Item(Dims) ≤ Bin(Dims).
- **Rigid Body Physics:** Ensures item dimensions match Master Data (no "morphing").
- **Collision Check:** Scans 3D coordinates to ensure no two bins physically overlap.

B. Data Integrity & Math

- **Utilization Audit:** Recalculates occupied volume vs. bin volume (1% tolerance).
- **Grid Consistency:** Validates internal math (Grid XYZ = MaxUnits).



C. Business Logic & Optimization

- **Inventory Reconciliation:** Compares Allocated Qty vs. On-Hand Demand.
- **Unallocated Logic:** If an item is unslotted, the system attempts to fit it into **every** empty bin type (using 6-axis rotation) to prove it physically couldn't fit.

4 Discussion and Analysis - Heuristic Results

INITIAL STATE

Weight Violations
0 (OK)

Misplaced A-Items
17 (!)

Avg Combined Score	261.0
Zone Reward	27.6
Util Reward	285.2
Dist Penalty	-51.8
Avg Util (Occupied)	35.65%
%Occupied Bins	48.7%

HEURISTIC OPTIMIZATION

FINAL OPTIMIZED STATE

Weight Violations
0 (OK)

Misplaced A-Items
14 (!)

Avg Combined Score	293.6
Zone Reward	54.0
Util Reward	273.3
Dist Penalty	-33.6
Avg Util (Occupied)	34.16%
%Occupied Bins	48.7%

The heuristic optimization was executed on a simulated warehouse environment containing 174 allocated items. The system performance was evaluated using the 'Average Combined Score' metric and validated against a secondary diagnostic tool to ensure zero physical violations.

4 Discussion and Analysis - RL Results

INITIAL STATE

Weight Violations
0 (OK)

Misplaced A-Items
17 (!)

Avg Combined Score	261.0
Zone Reward	27.6
Util Reward	285.2
Dist Penalty	-51.8
Avg Util (Occupied)	35.65%
%Occupied Bins	48.7%

RL OPTIMIZATION

RL OPTIMIZED STATE

Weight Violations
0 (OK)

Misplaced A-Items
0 (OK)

Avg Combined Score	441.5
Zone Reward	177.8
Util Reward	306.7
Dist Penalty	-43.0
Avg Util (Occupied)	38.33%
%Occupied Bins	50.4% (180/357)

The RL optimization was executed in a simulated warehouse environment containing 174 allocated items. The system performance was evaluated using the 'Average Combined Score' metric following episodes of box-level RL decision-making.

4 Discussion and Analysis - RL Results

Deeper Insight into RL Optimization

- **Balanced use of bins and space**
  The RL model increases the number of occupied bins slightly, but achieves higher average volume utilization per bin, avoiding both over-packing and under-use.

- **Avoids extreme strategies**
  Instead of minimizing bins at all costs or spreading items too thin, RL learns a middle ground that improves overall warehouse efficiency.

- **Utilization improves without congestion**
  Higher volume utilization is achieved without creating bottlenecks in fast or ergonomic zones, maintaining smooth operations.

- **Trade-off learned through rewards**
  The reward function encourages both space efficiency and accessibility, allowing RL to discover a practical balance rather than a rigid rule.

- **Result reflects real warehouse behavior**
  The final layout resembles how experienced planners balance space savings with operational flexibility, but is achieved automatically through learning.

4 Discussion and Analysis - Conclusion

1. Quantitative Dominance

- **Overall Score:** RL agent outperformed the Heuristic baseline by **+69.2%** (441.5 vs. 293.6).
- **Safety Compliance:** RL achieved **100% accuracy** (0 misplaced Class A items); Heuristic failed vertical safety constraints.

2. Algorithmic Trade-off

- **Heuristic (Greedy):** Prioritized linear **Velocity** (closest bin). Resulted in fragmented space and safety violations.
- **RL (Adaptive):** Prioritized **Multi-Objective Logic**. Accepted marginal travel costs to secure Density and Ergonomics.

3. Spatial Strategy

- **Demand Clustering:** RL learned to place high-demand items strictly in the **Ergonomic Zone** without hard-coded rules.
- **Smart Sprawl:** RL utilized slightly more bins (+1.7%) to achieve higher average fill rates (+2.7%).

4. Final Verdict

- **Static vs. Dynamic:** Rigid rules fail to reconcile conflicting goals (e.g., Speed vs. Safety).
- **Conclusion:** RL successfully solves non-linear trade-offs, offering a robust path for autonomous, safety-compliant WMS.

5 References

Duque-Jaramillo et al. (2024) – Sorting-Based Slotting Approach

- Proposes a structured slotting methodology integrating SKU demand, physical dimensions, and warehouse layout.
- Demonstrates that **ABC classification combined with structured slot assignment** significantly reduces total picking time and validates results using simulation with real operational data.

Contributions:

- Confirms **ABC classification as an effective baseline** for warehouse optimization.
- Supports the idea that **slotting decisions strongly affect picking efficiency**.
- Motivates the need for **simulation-based evaluation**, which is also used in this project.

5 References

Ju & Liu (2017) – Goods Location Optimization Based on Greedy Algorithm

- Applies a greedy algorithm to optimize item placement based on turnover rate.
- Uses distance-to-exit and demand frequency as primary optimization criteria.

Contributions:

- Provides a simple and interpretable heuristic baseline.
- Reinforces classical principles such as high-turnover items near exits.
- Serves as a conceptual reference for the heuristic comparator against RL.
- Highlights the limitations of greedy methods in handling complex constraints.

5 References

Mojumder & Nuruzzaman (2025) – AI-Driven Warehouse Optimization Review

- Conducts a large-scale systematic review of AI applications in warehouse optimization.
- Reports 15–45% efficiency gains and 20–35% improvements in space utilization using AI.
- Identifies Reinforcement Learning as particularly effective in dynamic, high-complexity environments.

Contributions:

- Provides theoretical and empirical justification for using RL.
- Reinforces the relevance of learning-based relocation instead of static slotting.
- Supports the idea that space utilization is often underrepresented in traditional optimization.

5 References

Cestero et al. (2022) – Storehouse: A Reinforcement Learning Environment for Optimizing Warehouse Management

- Warehouse modeled as a 2D grid world with entry and delivery points. It serves as the environment for RL agent to manage warehouse tasks (modelled as a Markov Process).
- Uses RL algorithms (DQN, PPO, A2C) and compares them against human-designed policies.
- Emphasizes adaptability and automation over static, handcrafted strategies.

Contributions:

- Confirms that RL is suitable decision-making problems involving sequential actions.
- Provides a conceptual foundation for modeling warehouses as RL environments.
- Motivates the use of simulation as a safe validation tool before real deployment.
- Shows the importance of invalid-action filtering, which aligns with your strict feasibility enforcement.

Thank you

Binil Sajeev, Abdelraheem Zekry, Orlando Manrique, Arthur Reis