Summer 6-21-2023

Designs of Puzzle Based Storage Systems with Unidimensional Cells

Hector J. Carlo
University of Puerto Rico - Mayaguez, hector.carlo@upr.edu

Andres Blanco
University of Puerto Rico - Mayaguez, andres.blanco1@upr.edu

Design of Puzzle-Based Storage Systems with Unidimensional Cells

Héctor J. Carlo
Industrial Engineering
University of Puerto Rico - Mayagüez
Mayagüez, Puerto Rico
ORCiD: 0000-0003-4297-085X

Andrés F. Blanco-Quintana
Industrial Engineering
University of Puerto Rico - Mayagüez
Mayagüez, Puerto Rico
ORCiD: 0009-0007-3710-1701

***Abstract*— A paradigm for puzzle-based storage systems (PBS) is that all cells have the capacity of transporting loads horizontally and vertically. This research challenges this paradigm by considering unidimensional PBS, where some cells are limited to either horizontal or vertical movements. PBS with some unidimensional cells are referred to as unidimensional PBS. A linear program (LP) formulation is presented to find the optimal load retrieval path using a single-escort in a unidimensional PBS. The LP is solved recursively to understand the tradeoffs of unidimensional designs for a 4×4 system. It is concluded that unidimensional PBS can be designed such that the capital investment cost to throughput tradeoff is favorable.**

Keywords*— *puzzle-based storage, high-density storage, automated warehouse, optimization

I. Introduction

Puzzle-Based Storage Systems (PBS) are very high-density parts-to-person storage systems for unit loads. The PBS design is based on an automated grid composed of cells capable of sliding loads in the horizontal and vertical dimensions. Retrieving a load from a PBS requires sequentially moving the load to an open (empty) cell until reaching the output point. Since loads need to be slid into open cells, the open cells are commonly referred to as *escorts*.

Consider the 4×4 PBS grid depicted in Fig. 1a. The sixteen cells in the PBS either hold a load or are open (escort). This example has a single escort (colored white) that is initially located at the input/output (I/O) point on the southwest corner with coordinates (1,1). The requested load is represented by a black cell and is located at coordinate (3,3). The remaining fourteen cells, identified in gray, contain other stored loads. To move the desired load (black) closer to the I/O requires placing the escort in a cell either immediately west or south of the desired load. Although technically the escort does not move (as the escort is an empty cell so there is nothing to move), it is easier to refer to "moving the escort," which occurs when moving a load to an open cell. In this example, as illustrated by the arrows in Fig. 1(a), the escort is to be moved to the cell south of the desired load by moving the loads originally in cells (2,1) and (3,1) simultaneously west and then moving the load in cell (3,2) south. This type of movement is known in the literature as block or tandem movements. Notice that it is not possible to move the load in cell (3,2) south at the same time the load in that cell is moving west due to interference. After moving the desired load south, the escort is now the desired load's original location with coordinate (3,3), as in Fig. 1(b). The remaining of Fig. 1 depicts the sequence of cell movements required to retrieve the desired load.

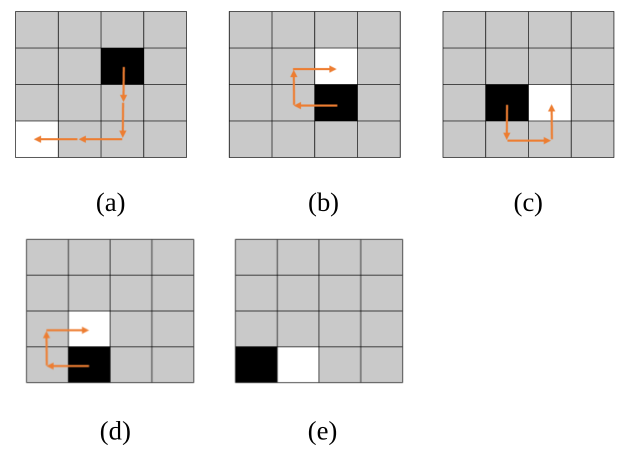

Fig. 1. Sequence of cell movements to retrieve a load with a single escort.

Two main PBS designs are described in the literature: *pop-up* and *shuttle*. In the pop-up design, every cell has a conveyor for sliding loads in one dimension and a pop-up conveyor to achieve orthogonal movements (e.g., a roller conveyor with pop-up rollers). On the other hand, the shuttle design involves storing loads on a dedicated material handling equipment (MHE) that moves horizontally or vertically to an open space. Alternatively, the shuttle system may use a MHE to pick-up and move loads. This study focuses on PBS with a pop-up design. Pop-up designs achieve much faster retrieval speeds at the expense of a more rigid configuration (e.g., fixed grid shape and cell size).

The main implementation challenge for PBS is the investment (capital) cost. It is observed that reducing the number of pop-up conveyors would simultaneously reduce the investment cost and operational complexity of PBS. Hence, this study challenges the premise that every cell in a PBS must have a pop-up conveyor. The tradeoff of eliminating a pop-up conveyor is that load movements would be limited to one dimension, which could increase the expected retrieval time for a load. Cells without a pop-up conveyor are henceforth referred to as *unidimensional cells* since movements are limited to bidirectional travel in only one dimension. A PBS that includes unidimensional cells is termed *unidimensional PBS*.

This research seeks to develop insights for designing unidimensional PBS with a single escort. The remaining of this paper is organized as follows. Section II summarizes the academic literature associated with mathematical models of PBS. Section III describes the problem, presents the modelling assumptions, and proposes a linear program (LP) formulation to optimize the retrieval sequence for a load. Experimental results in Section IV start with a bi-dimensional system and solves the LP iteratively to identify the cell has the least impact to the expected load retrieval time when becoming unidimensional. Lastly, Section V presents the conclusions and future work.

II. Literature Review

Most of the models developed in the existing literature assume that cells may move in both dimensions. These models are based on some movements that might not be feasible in unidimensional PBS. Gue & Kim [1] implicitly assume that there is only one cell moving in the system at a time. Hence, an escort moves progressively from one cell to the next. Under this assumption, if the load just completed a movement, it would take the escort 3 or 5 movements to move the load again. Clearly, if some cells are unidimensional, the existence of a 3- or 5-move cannot be guaranteed. Papers that optimize the retrieval sequence for a PBS by moving one cell at a time include [1]-[4]. There are other publications that allow tandem movements for the escort, which is able to move the load with 3- and 4-moves instead [4]-[6].

The only published work that implicitly assumes unidimensional PBS is [7], which modifies the decentralized adapted controls algorithm for *GridFlow 2D* storage system from [8] – which is composed of modular *FlexConveyors* – so it can react to conveyor failures. The result is a control algorithm that enables the system to continue operating even if some north-south or east-west conveyors fail. The authors report, as an example, that based on an agent-based simulation two failed north-south conveyor failures in the same row have a significantly greater impact on the throughput than two north-south conveyor failures in the same column. The main difference between the work of [7] and this study is that they present a modification of the controls system to cope with unreliable conveyors, whereas this study seeks to optimize the placement of unidimensional cells in a PBS.

III. Problem Description

Consider an $m \times n$ PBS grid with a single escort. The problem under study seeks to determine the best path to retrieve a predetermined load in the least amount of time (measured in number of movements) by means of horizontal and vertical movements.

A. Modelling Assumptions

The following modelling assumptions are made:

- The initial location of the requested item is known
- There is only one escort
- There is only one I/O point
- The travel distance between locations is known and corresponds to rectilinear travel considering unidimensional cells
- Each horizontal or vertical movement is of one distance unit (DU)
- Loads may only move to the escort

B. Problem Formulation

This Section presents an LP formulation for determining the best route to retrieve a predetermined load from its initial location to the I/O in the least amount of time using a single escort. The following nomenclature is defined:

Sets:

$\boldsymbol{N}$ = set of all (grid) cell locations, $\boldsymbol{N} = \{1, \dots, \|\boldsymbol{N}\|\}$

$\boldsymbol{J_k}$ = set of neighborhood locations for cell location k, $\boldsymbol{J_k} = \{1, \dots, \|\boldsymbol{J_k}\|\}, k \in \boldsymbol{N}$

Indices:

i = starting location of escort for a movement, $i \in \boldsymbol{N}$

k = end location of the escort for a movement; it is also the location of the load to be retrieved, $k \in \boldsymbol{N}$

j = last cell location visited by escort before reaching k, $j \in \boldsymbol{J_k}$

Parameters:

s = initial location of the load to be retrieved, $s \in \boldsymbol{N}$

r = location of the I/O, $r \in \boldsymbol{N}$

d_{js} = minimum distance for the escort to move from its initial location to s through neighborhood location j in DUs

d_{ijk} = minimum distance for the escort to move from location i, using the intermediate location j, to reach the load located at k. This distance cannot go directly from location i to location k as this would imply that the load is moved in the process.

Decision Variables:

y_j = Binary variable equal to 1 if escort moves from its initial location to s through j, zero otherwise. This variable is relaxed in the formulation.

x_{ijk} = Binary variable equal to 1 if escort moves from location i to the item's location k via intermediate location j, zero otherwise. This variable is relaxed in the formulation.

$$Min\ z = \sum_{j \in \boldsymbol{J_s}} d_{js}\, y_j + \sum_{i \in \boldsymbol{N}} \sum_{j \in \boldsymbol{J_k}} \sum_{k \in \boldsymbol{N}} d_{ijk}\, x_{ijk} \qquad (1)$$

s.t.

$$\sum_{j \in \boldsymbol{J_s}} y_j = 1 \qquad (2)$$

$$\sum_{h \in \boldsymbol{J_j}} x_{shj} = y_j \ \forall j \in \boldsymbol{J_s} \qquad (3)$$

$$\sum_{i \in \boldsymbol{N}} x_{ijk} = \sum_{h \in \boldsymbol{J_k}} x_{khj} + \sum_{i \in \boldsymbol{N}} x_{irk} \ \ \forall k \in \boldsymbol{N} \backslash \{s\}, j \in \boldsymbol{J_k} \qquad (4)$$

$$\sum_{i \in \boldsymbol{N}} \sum_{k \in \boldsymbol{N}} x_{irk} + y_r = 1 \qquad (5)$$

$$0 \leq y_j \leq 1 \ \ \forall j \in \boldsymbol{J_s} \tag{6}$$

$$0 \leq x_{ijk} \leq 1 \ \ \forall i, k \in \boldsymbol{N}, j \in \boldsymbol{J_k} \tag{7}$$

Objective function (1) minimizes the distance from the escort's initial location to the load's initial location and all the escort movements from there on until reaching the I/O. Since an escort movement implies a load movement, the total time required to retrieve a load is directly related to the number of movements required by the escort. The model can consider the case where cell movement is limited to one at a time or can allow tandem movements by manipulating the distance matrix. Constraint set (2) ensures that the escort moves from its initial location exactly once. The initial location of the escort is not explicitly specified but is implicitly included in d_{js}. It is noted that if the escort travels to any location via the I/O, the load will end up at the I/O, which concludes the retrieval process. Constraint set (3) ensures the escort repositions for the second load movement, which starts at the escort's ending location after the first load movement (i.e., the load's initial location) and ends at the intermediate location used to reach s in the load's first movement. Constraint set (4) is a flow conservation constraint that ensures that if the escort moves a load at location k, the next escort movement needs to start at k or the load must have been delivered at the I/O (i.e., intermediate location r was visited). Constraint set (5) covers a special case where the load is delivered to the I/O on the first move. Constraint sets (6) and (7) bound the decision variables. Intuitively, the variables should be binary; however, since the coefficients associated with the restrictions create a Totally Unimodular matrix, the variables may be relaxed.

IV. Experimental Results

The formulation in Section III.B assumes that the unidimensional PBS layout is given. The LP formulation may be used to answer the following research questions:

Q1: What is the tradeoff in expected retrieval times if some cells are unidimensional?

Q2: How should unidimensional PBS be designed?

A Python code that uses Gurobi 9.1.2 to solve the proposed LP formulation was created. Experiments consider a 4×4 PBS grid. The code starts with a bi-dimensional PBS and the desired load in one of the cells. The code solves the LP 15 times, each time with a different initial location of the escort. The average of these results represents the expected retrieval distance for a load in that particular location. This process is repeated considering the load in each of the cells ($15 \times 15 = 225$ scenarios). The average of the 225 scenarios represents the expected optimal retrieval distance for the PBS layout considered (bi-dimensional PBS in this case), considering a load and escort in a random location. The strategy described can be used to determine the expected optimal retrieval distance for any PBS layout.

For the experiments, the expected optimal retrieval distance is computed considering one unidimensional cell at a time (i.e., have one of the cells be limited to horizontal or vertical movements). The four corners of the grid must be bidimensional as a unidimensional corner would gridlock the escort when retrieving a load from the corner cell. In such a case, the only alternative would be to move the escort to the neighboring location, but that would move the load back into the corner.

In the 4×4 PBS grid there are 12 cells that may be unidimensional (four corners excluded), and the unidimensional cells may be in one of two orientations, so a total of 24 layouts are considered. The expected optimal retrieval distance for the 24 layouts is compared in terms of the additional distance. The unidimensional cell (and orientation) with the shortest expected optimal retrieval distance is considered the best one as it has the least impact in terms of the PBS throughput.

Assuming that the unidimensional cell with the minimum expected optimal retrieval distance is implemented, the strategy is also used to determine the second unidimensional cell to be implemented. Clearly, a particular cell may not be unidimensional in both orientations as that would prevent any movement through that cell. Hence, only 22 unidimensional cells are considered (corners and the first unidimensional cell are excluded). After the second unidimensional cell was selected, a third unidimensional cell was identified, and so on. Fig. 2 summarizes the results expected optimal retrieval distance as a function of the number of unidimensional cells incorporated. The cost of the PBS will be proportional to the number of unidimensional cells. Clearly, the proposed strategy that incorporates one unidimensional cell at a time does not guarantee optimality of the resulting layout.

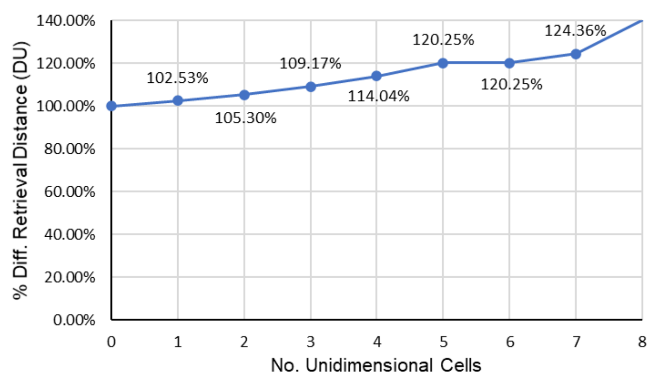

Fig. 2. Tradeoff between number of unidimensional cells and expected retrieval distance.

It can be observed from Fig. 2 that making 25% of the cells unidimensional (3/12 as the corners need to be bi-dimensional) affects the expected optimal retrieval distance by less than 10%, when compared to the bidimensional PBS. Considering four unidimensional cells affects the expected optimal retrieval distance by less than 15%. With six unidimensional cells the expected optimal retrieval distance increases by 20%.

Fig. 3 depicts the final layout with as many unidimensional cells as possible. The bi-dimensional grids are presented in gray and the unidimensional grids only include the allowable move direction. The number included in the unidimensional cells indicate at which stage it was incorporated (1 being the first unidimensional cell added). This number also corresponds to the abscissa in Fig. 2. The maximum number of unidimensional cells is 8. Adding a ninth unidimensional cell renders the layout infeasible as the I/O would be unreachable from cells. There were instances where two unidimensional cells resulted in the same expected retrieval distance; in those cases, the more intuitive choice was selected by hand. In all these cases the unidimensional cell not selected was selected in the next stage.

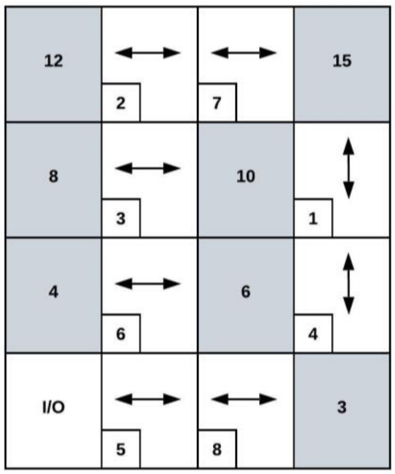

Fig. 3. Unidimensional PBS layout.

The unidimensional PBS layout in Fig. 3 appears to have created North-South corridors in the first and last columns and East-West corridors in the other columns. This layout may be used to guide the design of unidimensional PBS considering the tradeoff between cost and expected retrieval distance from Fig. 2.

Larger PBS grids would be expected to follow the same pattern; edge cells would become unidimensional and corridors will be created for loads and escorts to navigate. If more escorts are incorporated the layout could change significantly as corner cells may become unidimensional.

V. Conclusions and Future Work

This research studies puzzle-based systems (PBS) where some cells are limited to horizontal or vertical movements. PBS with some unidimensional cells are referred to as unidimensional PBS. This study presents a linear program (LP) formulation to find the optimal load retrieval path using a single-escort in a unidimensional PBS. The LP is solved recursively to determine the expected optimal retrieval distance for a layout considering a random load and escort location for a 4×4 PBS system. The unidimensional cell with the least impact on the expected optimal retrieval distance is incorporated into the design, and the process is repeated to progressively determine the next unidimensional cells to incorporate. The maximum number of unidimensional cells that could be implemented were 8 (out of the 12 cells, excluding the corners). It is concluded that a layout including 3 unidimensional cells affected the expected optimal retrieval distance by less than 10%, when compared to the bidimensional PBS. Adding four unidimensional cells affected the expected optimal retrieval distance by less than 15%. Considering six unidimensional cells further increased the expected optimal retrieval distance by less than 20%.

Since the unidimensional cells were added one-by-one, the resulting unidimensional PBS layout is not guaranteed to be optimal. However, the results suggest that unidimensional PBS can be designed such that the capital investment cost to expected retrieval distance (associated with PBS throughput) tradeoff is favorable.

Future work can extend this research to consider multiple escorts, alternative I/O positions, and to incorporate replenishment operations.

References

[1] Gue, K.R. and B.S. Kim, "Puzzle-based storage systems," Naval Research Logistics, vol. 54(5), pp. 556-567, 2007.

[2] Kota, V.R., Taylor, D., and K.R., Gue, "Retrieval time performance in puzzle-based storage systems," Journal of Manufacturing Technology Management, vol. 26(4), pp. 582 – 602, 2015.

[3] Mirzaei, M., De Koster, R., and N. Zaerpour, "Modelling load retrievals in puzzle-based storage systems," International Journal of Production Research, vol. 55(21), pp. 6423-6435, 2017.

[4] Raviv, T., Bukchin, Y., and R. de Koster, "Optimal retrieval in Puzzle-Based Storage systems using automated mobile robots," Transportation Science, vol. 57(2), pp. 424-443, 2023.

[5] Bukchin, Y., and T. Raviv, "A comprehensive toolbox for load retrieval in puzzle-based storage systems with simultaneous movements," Transportation Research Part B: Methodological, vol. 166, pp. 348-373, 2022.

[6] Yu, H., Yu, Y., and R. de Koster, "Optimal algorithms for scheduling multiple simultaneously movable empty cells to retrieve a load in puzzle-based storage systems," Available at SSRN: https://ssrn.com/abstract=3506480, 2016.

[7] Furmans, K., Gue, K.R., and Z. Seibold, "Optimization of failure behavior of a decentralized high-density 2D storage system," In Dynamics in Logistics: Third International Conference, LDIC 2012 Bremen, Germany, February/March 2012 Proceedings, pp. 415-425. Springer Berlin Heidelberg, 2013.

[8] Gue K.R. and K. Furmans, "Decentralized control in a grid-based storage system," Proceedings of the 2011 Industrial Engineering Research Conference, eds. T. Doolen and E. Van Aken, 2011.