Mixed Pallet Stacking: An Overview and Summary

Stephen Balakirsky
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD USA
stephen@nist.gov

Henrik I. Christensen
RIM@GT,CoC
Georgia Institute of Technology
Atlanta, GA USA
hic@cc.gatech.edu

Thomas Kramer
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD USA
thomas.kramer@nist.gov

Pushkar Kolhe
RIM@GT,CoC
Georgia Institute of Technology
Atlanta, GA USA
pushkar@cc.gatech.edu

Frederick Proctor
National Institute of Standards and Technology
100 Bureau Drive
Gaithersburg, MD USA
frederick.proctor@nist.gov

ABSTRACT

Stacking boxes of various sizes and contents on pallets (i.e. making mixed pallets) is a primary method of preparing goods for shipment from a warehouse to a store or other distant site. A special session of the 2010 PERMIS workshop was held to examine mixed palletizing issues. Papers were presented by end-users, vendors, researchers, and evaluators of palletizing solutions. This paper presents a summary of this session and discusses issues ranging from how to construct pallet build plans to metrics for evaluating these plans to competitions that feature novel palletizing approaches.

Categories and Subject Descriptors

I.2.1 [**Computing Methodologies**]: Artificial IntelligenceApplications and Expert Systems; D.2.8 [**Software**]: Software EngineeringMetrics

General Terms

Algorithms, Performance

Keywords

Palletizing, Simulation, Metrics, Mixed, Pallet

1. INTRODUCTION

The warehousing and distribution industry is an important part of the supply chain that transports goods from the factory to the end user. A typical scenario is receiving pallets containing a single Stock Keeping Unit (SKU), depalletizing the product, and then creating new mixed SKU pallets for transport to customers. About 100,000 containers per week are imported into the US each day. Of these, about 30% of the containers are immediately repacked onto mixed pallets for distribution to stores. In the grocery industry alone, more than 80 million SKUs (half of which are for beverages) are distributed to grocery stores each week. Much of this work is performed as a physically demanding manual task which has the largest labor costs, greatest potential for product damage, and most prevalent occurrence of order inaccuracies [5] of an activity in the distribution process. Increasing numbers of SKU's coupled with the manual nature of the task is causing a disruption in this industry due to increased costs from product damage and fines/penalties for shipping inaccuracies [5].

There is thus no doubt that a major challenge for supply chain management and logistics is optimization of mixed palletizing systems. At the PERMIS session, it was estimated by C&S Wholesale Grocers that the manufacturing costs only represent 30% of the consumer costs for grocery items. The remaining costs are related to distribution, handling, and sales. Optimization of the process has the potential to reduce consumer prices significantly. Applying robotics to this problem has the potential to greatly reduce both product damage and order inaccuracies, thus further reducing consumer prices.

The palletizing robotic work-cell may be seen as an automation center that has an in-feed conveyor for product coming into the cell, a robotic arm with end-of-arm tooling and planning software, and an out-feed conveyor for carrying the mixed case pallets from the robot cell to a distribution center. There are multiple providers of logistic systems and complete solutions for distribution centers. This includes methods for depalletizing, warehouse managements, order management, palletizing algorithms, gripper design, robot automation, mixed palletizing and infrastructure support such as conveyers or AGV systems. However, while several companies already provide solutions for this problem, there is no consistent world view on what makes a good pallet, nor standards for the information models required to complete this task.

The 2010 Performance Metrics for Intelligent Systems workshop featured a special session on mixed palletizing that addressed these deficiencies. This session brought together representatives from the end-user community, robot vendors, and researchers to share and compare ideas on the current

PERMIS' 10 September 28-30, 2010, Baltimore, MD USA

state and future challenges of automation for mixed palletizing. Several papers and talks were presented, and a panel discussion was held. The papers included an end-user perspective of the palletizing problem [10], a description of the state-of-the-art and continuing challenges for robotic palletizintg systems [5], a survey of current planning systems for mixed palletizing [6], and a discussion of emerging information standards and benchmarks for palletizing metrics and evaluation tools [1]. In this paper we summarize this session by presenting an overview of the generic needs of a palletizing solution and some present palletizing systems, planning systems for palletizing and evaluation benchmarks.

Section 2 of this paper presents an overview of current pallet planning algorithms. A summary of the discussion on the information model is next presented in Section 3, followed by a summary of the metrics discussion in Section 4. Finally, information on a competition that is striving to advance the state-of-the-art in palletizing is presented in Section 5 and conclusions are presented in Section 6.

2. PALLET PLANNING TECHNIQUES

Mixed palletizing is a multi-objective multi-constraint optimization problem.

There are at least two different common modes for the way in which boxes are supported by other boxes. In one mode, which we might call the *cardboard* mode, the structure of the box bears the weight of any item placed on top of the box. *Cardboard* mode occurs with boxes containing breakable items such as glassware and tomatoes. These boxes may be made of more sturdy material such as wood or thick plastic or may have additional internal supports such as columns at the corners. In the other mode, which we might call *contents* mode, it is the contents of the box that provides most of the support for any weight placed on top of the box. *Contents* mode occurs with boxes containing relatively robust items such as cans of soda or reams of paper. In the *contents* mode, the material from which the box is made may have very little resistance to compression. The contents have it, instead.

The planning system must be able to vary its strategy depending on the support structure of the boxes. When building a stack of boxes in *cardboard* mode, overlap (a box resting on multiple boxes beneath it) is usually a bad thing because the edges of the boxes (the load-bearing part) are not lined up vertically. One web site [4] states that overlap (also called interlocking) “can destroy up to 50% of the compression strength”. In *contents* mode, overlap is generally good, since it tends to hold the stack together. This is not an item for optimization, this is a hard constraint that the planner must be able to adapt to.

Other hard constraints on the planning system deal with the amount of *a priori* data available to the planner. In off-line planning systems, the planner is able to not only specify the final resting location of each box, but is also able to plan the order in which the boxes are placed on the infeed conveyor. In real-time planning systems, the planner does not have *a priori* knowledge of the box ordering and must compute a location for each box as it is received. The remained of this paper will deal with the contents mode offline planning systems and will include creation of pallets with single or multiple SKUs.

In the literature, the mixed palletizing domain for which we are developing metrics is often called “the distributor’s pallet packing problem”. It is one of a set of closely-related packing (or unpacking/cutting) problems, all of which are known to be hard to solve as pure geometry problems. In fact, the pallet stacking problem is related to the bin packing problem which is a standard NP-hard (non-deterministic polynomial-time hard) problem in computational complexity theory.

In this problem, objects of different volumes must be packed into a finite number of bins of limited capacity in a way that minimizes the number of bins used. By restricting the number of bins to one (our one pallet), and characterizing each item by both a volume and a value, we have a problem known as the knapsack problem. Our palletizing problem is then a special case of the knapsack problem where each box is represented by a volume/weight and a potential profit (the value) for including that box on the pallet. Consider a knapsack with item set N, consisting on n items, where the j_{th} item has profit p_j and weight w_j, and the capacity is c. Then, the objective function can be formulated as:

$$\max \sum_{j=1}^{n} p_j x_j \tag{1}$$

subject to

$$\max \sum_{j=1}^{n} w_j x_j \leq c \tag{2}$$

$$x_j \in \{0, 1\},\ j = 1, \ldots, n. \tag{3}$$

In equation (2) x_j can only take integral values and it denotes whether the j_{th} item is included in the knapsack or not. Finding the optimum solution vector x^* having an optimum profit z^* is non-trivial and known to be NP-hard. There are 2 main aspects of the palletizing problem that need to be defined: the constraints and the model.

2.1 Planning Constraints

2.1.1 Stability Constraints

The stability constraints of a mixed palletizing problem can be specified by studying the shear and stress properties and doing a finite element analysis (FEM) on the pallet. Stress and strain failure are caused by the load or pressure on each SKU. The planner has to take these into account for scheduling the order in which boxes should be packed.

These constraints are formally expressed as compressive or shear stress and yield equations. Compressive stress causes the package to reduce its height due to pressure and buckle. The strength of the pallet is determined by the elastic properties of the boxes it contains. This property allows us to constrain the maximum allowable pressure on each box by (4). The stress strain curve is a standard tool used in FEM analysis which gives (5). It puts a limit on the assumed yield strength of a package.

$$\rho = \frac{F}{A} \propto \frac{\pi^2 E}{h^2}\ . \tag{4}$$

$$\partial\rho \leq k\frac{dh}{h}\ . \tag{5}$$

Here E is the Young’s modulus and h is the height of the package, a constant of elasticity specific to the packaging material being used. F, A are the total force and the total surface area of the package. From (4) we can determine that shear stress can be reduced by creating interlocking patterns which would increase the area under contact for each package. A trivial approach to model this would be to maximize,

$$P \propto \frac{nA_0}{A}, \tag{6}$$

where n is the total number of boxes touching the current box and A_0 is the total area encompassed by all the boxes. We normalize it by A, which is the total surface area of the box.

2.1.2 Packaging Constraints

Finding a feasible packing for a pallet is referred to as the cutting problem. There are two widely studied cutting problems: guillotine and non-guillotine cutting problems. Guillotine patterns refer to patterns that are cuttable. The problem we consider in this section is that of selecting a subset of items and assigning coordinates (x_j, y_j, z_j) to each item, such that no item goes outside the bin, no two items overlap and the total volume of the items does not exceed the maximum capacity. We have the following constraints,

$$\begin{aligned} 0 \leq x_i &\leq W - w_i, \\ 0 \leq y_i &\leq H - h_i, \\ 0 \leq z_i &\leq D - d_i. \end{aligned}$$

To ensure that no two packed boxes i, j overlap we add more constraints,

$$\begin{aligned} x_i + w_i &\leq x_j, \\ y_i + h_i &\leq y_j, \\ z_i + d_i &\leq z_j, \\ x_j + w_j &\leq x_i, \\ y_j + h_j &\leq y_i, \\ z_j + d_j &\leq z_i. \end{aligned}$$

Many methods that solve the bin packing problem use the above constraints.

2.1.3 Empirical Constraints

Empirical constraints constitute the generic statistics that are known to be related to stability such as the center of mass and density of the pallet.

2.2 Planning Models

2.2.1 Variations of the Packing Problem

Variations of the packing problem include knapsack loading, container loading, and bin packing. In knapsack loading, the problem is to find a subset of items that will fit into a single bin. In container loading, the problem is to find a feasible arrangement of items in which the height of the filled bin is minimum, and in bin packing, the items are packed into finite sized bins and the problem is to find the solution with the minimum number of bins. Usually one of these problem determines a cost function for the knapsack problem.

Any logistics or a mixed palletizing system can be modeled by a Knapsack problem variant. If the planning problem involves many constraints then we can frame it as a *d-dimensional knapsack problem.* Planning for rainbow palletizing (a pallet where different layers of the pallet are created out of different products) can be performed by modifying the problem as a *quadratic knapsack problem.* Other interesting variants of the problem can be used to model complex warehousing systems for shipping industries. The *multiple knapsack problem* and the *multiple-choice knapsack problem* can be used to model package priority, packages arriving late or even wrongly delivered packages. If the rate or the number of packages coming into a warehouse is not fixed, a variant called *stochastic knapsack problem* can be used for modeling on-line bin packing problem. All these different models can give us insight into the measure of efficiency a logistics scheme is capable of. There are lower bounds available from theoretical computer science, which give a theoretical indication that the running time of an exact algorithm for the knapsack problem can not beat a certain threshold under reasonable assumptions on the model of computation. Many knapsack instances can be solved within reasonable time by exact solution methods.

2.2.2 Planning Algorithms and Benchmarks

Algorithms that solve NP-hard problems can be divided into two parts: *exact algorithms* and *approximate algorithms.* Exact algorithms find the most optimal solution of a given problem whereas approximate algorithms find an approximate solution.

There are many standard algorithmic approaches that can be used to solve this problem. Greedy, branch and bound, dynamic programming and polynomial time approximation schemes are used to solve palletizing problems modeled as the knapsack problem. Each of these different class of algorithms scale the profit or weight space to achieve a near optimal solution. For problems with stochastic input, heuristics are used which are learned from statistical distribution techniques to model the profit and weight space.

2.2.3 Algorithmic Benchmarks

Algorithms that solve the knapsack problem are not similar. Algorithms that find the most optimum solution often do so by doing a exhaustive search. They are computationally very inefficient when compared to approximate algorithms which are computationally far more efficient. However, approximate algorithms have the drawback that they can only find a near optimum solution. It is not always possible to do an exhaustive search over a problem space.

Algorithms are usually tested over several data sets and their performance is determined analytically. One of the most common way to check is to compare running time of the algorithm after the data set is doubled. The time required to find the solution should not increase exponentially for polynomial or pseudo polynomial algorithms. The most common way to measure the performance of an algorithm is to perform the worst-case analysis.

2.3 An Exact Approach

Consider the robot packable bin packing problem. The problem is strongly NP-hard. However we discuss here an exact branch and bound algorithm for solving the bin packing problem. The three dimensional bin packing problem

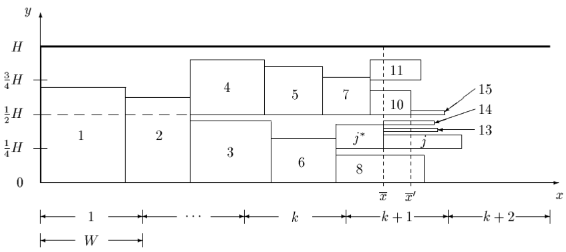

Figure 1: The shelf approach used for packing boxes in shelves [8]. A free box is assigned to the same shelf or a new shelf is created.

consists of orthogonally packing all the boxes in the minimum number of bins.

A greedy approach to solve this problem would assume that the packages are sorted according to decreasing efficiencies

$$\frac{p_1}{w_1} \geq \frac{p_2}{w_2} \geq \cdots \geq \frac{p_n}{w_n} \ . \tag{7}$$

An obvious disadvantage of this approach is that only the first bin would be efficiently filled. The exact algorithm due to [8, 7] is based on a two level decomposition principle: the main branching algorithm and the single bin filling algorithm. The main branching algorithm assigns boxes to each bin, while a separate branch and bound algorithm tests whether the box can be placed in the bin by determining its location and cost. The main branching algorithm is an approximation algorithm and limits the number of execution of the single bin filling algorithm. The approximation algorithm can be defined by many complementary heuristics.

Consider our objective function is to maximize the number of boxes to fit in a bin. Let Z be the solution value and let $M = \{1, 2, \ldots, m\}$ be the current set of bins to which the boxes are assigned.

2.3.1 The main branching algorithm

The exploration of the main branching algorithm follows a depth-first strategy. At each decision node the free box is assigned to all open bins. It is also assigned to a new bin if $|M| < Z - 1$. Then the two heuristics on the single bin filling algorithm are used to find feasible packaging combinations. A node is not explored further if the lower bound is more than one or no feasible solution is found with the single bin filling algorithm. A box is assigned to the bin with the maximum upper bound.

2.3.2 The single bin filling algorithm

Here we consider two bin packing heuristics which have a complementary behavior. A poor performance on one of them corresponds to a good performance on the other.

In Figure 2.3.2, boxes are sorted according to their heights. All the boxes which lie within $\{\frac{h}{2^i}, \frac{h}{2^{i+1}}\}$ are put in group i. For every group i, boxes are sorted in slices of their width and length. If a new free box cannot be assigned to a particular shelf, it is put on the next shelf which starts from the highest box in the current shelf.

In Figure 2.3.2, is the 3D corners algorithm which decides feasible location after every assignment based on geometry. The packaging constraint equations in Section 2.1.2 correspond to this particular heuristic.

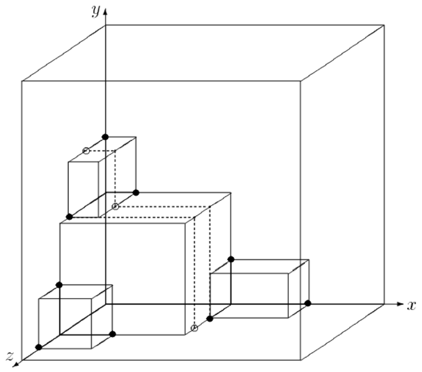

Figure 2: The 3D corners approach is used to find all possible feasible positions for a box [7]. The black dots indicate all starting positions available for a free box.

If a box cannot be assigned using one of the above heuristics it is rejected.

3. INFORMATION MODEL

An information model provides a sharable, stable,and organized structure of information for representing domain specific data. For the case of mixed palletizing, there are three categories of data that need to be represented. These include *a priori* data about possible articles and constraints, plan data that specifies how a pallet is to be constructed, and as-built data that provides a description of the pallet as it was actually constructed. In our case, all of these information models are provided as XML schemas. The *a priori* and plan models are based on extensions of KUKA Robotics'[1] information models. The as-built model is a newly created construct.

3.1 A priori information model

The *a priori* information model contains all of the data necessary for the planner to construct and optimize its plans. It contains information on pallet constraints, individual articles, and the bill of lading. Pallet constraints include information such as the pallet dimensions, allowed overhang, and maximum capacity. Article information includes quantities such as the article dimensions, weight, robustness, and family group.[2] Finally, the bill of lading contains order information such as the desired quantities for each article.

3.2 Plan information model

The plan information model contains all of the information necessary to construct the pallet. This includes article information, plan-build information, and article approach information. The article information contains the same data

---

[1] Certain commercial software tools and hardware are identified in this paper in order to explain our research. Such identification does not imply recommendation or endorsement by the authors, nor does it imply that the software tools and hardware identified are necessarily the best available for the purpose.

[2] A family group is a construct that is established to cluster items that are often sold together or items with specific storage or handling requirements. These items should be co-located on the final pallet.

as in the *a priori* information model described above with the addition of the pallet build order. The build order enumerates the order which articles should be placed on a conveyor system that feeds the robot arm that is constructing the actual pallet.

3.3 As-built information model

The as-built information model contains information on the pallet as it was actually constructed by the robotic platform (either simulated or in reality). The file contains the actual 6-degree-of-freedom location of article after completion of the build as well as information on the actual article size. The reason that size information is necessary is that packages are deformable. Their shape may change as pressure is applied to them from packages placed on top, or due to damage from the manipulation task. In addition, the actual packages used in the pallet build may differ from the expected models. At the current time, the packages used in simulation are not deformable, and are of uniform size and shape. Therefore, the information model contains an identifier for each package. The identifier may be used to obtain more information about the package from the order file. However, this is an area that we hope to expand upon in our future work.

4. PALLET METRICS

Due to the fact that planning for a mixed pallet is NP-hard, we know that computed plans will be suboptimal. We further know that packing strategy depends on the type of packages being stacked and pallet utilization depends on shipping constraints. However, given a packing plan, we would still like to answer the question of *how good is the plan?* In order to answer this question, metrics must be devised. Neither papers on pallet packing found in the literature nor the commercial systems say much about the metrics themselves, other than packing density. Brief mention is made in [2] of intersection, overlap, overhang, and center of gravity. In an attempt to better answer this question, the authors have developed a series of metrics that may be decomposed into the categories of Article Specific Metrics, Pallet-Wide Metrics, and Error Metrics.

4.1 Article Specific Metrics

Article specific metrics are those metrics which apply to a single article or package. They include measures that contribute to the stability (e.g. connections below and overlap) and integrity (e.g. overhang and maximum pressure) of individual packages.

The connections below measures the number of packages on which this package rests. This measure is designed to indicate the degree of interlocking that occurs on the pallet, where more interlocking may imply improved pallet stability. It should however be noted that for *cardboard mode* pallets, interlocking may significantly reduce the load capacity of individual articles. The overlap of a package represents the fraction of the bottom of the article that rests on other packages. Small values would mean poorly supported articles which could result in pallet instability.

The overhang indicates the amount of the article that extends over the pallet edge. Large overhang could make the article vulnerable to to damage, and may make loading the pallet into a truck, plane, or container difficult or impossible. The maximum pressure indicates the maximum static pressure that will be exerted on this package. Note that dynamic loads, for example when a package is dropped on top of this article, may cause the article to experience larger pressures.

4.2 Pallet-Wide Metrics

Pallet-wide metrics are those metrics which apply to the pallet when taken as a whole. They include measures that relate to the raw construction data for the pallet such as the number of articles on the pallet and overall pallet statistics such as the pallet's total weight, maximal height, and volume. In addition, efficiency and handling characteristics are represented with these metrics.

Efficiency measures strive to measure how efficient the pallet construction is. They include storage volume and volume density. The storage volume represents the warehouse space that is required to store the pallet based on the calculation of the pallet base area times its height. Volume density is a ratio that compares the sum of the individual article volumes to the storage volume. It may be used to indicate how much "empty space" is required to be stored.

Handling measures indicate how easy it will be for the final pallet to be maneuvered during shipping and the complexity of unloading the pallet. It includes such measures as the pallet's center of gravity and the number of product families that are contained on the pallet. A product family is a set of articles that will be unloaded at a single location at the pallet's destination. In the case of product families, a particular product family's distribution on the pallet is also considered. For example, a pallet that contains three layers of articles with layer one and three belonging to the same family would be said to contain three different families. The reason for this is that the pallet would need to make three stops to unload (stop one for the family in layer three, stop two for the family in layer two, and then back to stop one for layer one).

4.3 Error Metrics

The error metrics deal with problems in the packing plan. These problems include plan errors, stocking errors, and syntax errors. Plan errors occur when some planning constraint has been violated. These include constraints on article placement (e.g. two articles intersect) as well as constraints on article integrity such as the maximum pressure that may be exerted on an article and pallet integrity such as the overall pallet weight, height, and overhang.

Stocking errors occur when the content of the pallet does not match the order. This can include missing or extra articles. Finally, syntax errors occur when the plan file does not properly conform to the predefined XML schema for the plan.

4.4 Metric Tools

The above metrics define a set of raw values that can be computed from a combination of the raw order data, plan constraints, and final packing plan. In order to dictate that one packing plan is superior to another, these raw values must be combined and weighted into a single value. This weighting may be application and industry specific. In order to experiment with various weightings and examine the specific values of individual metrics, a tool has been created which computes these metrics. This tools is called the *Pallet Viewer* and is a C++ program that uses OpenGL graphics[9].

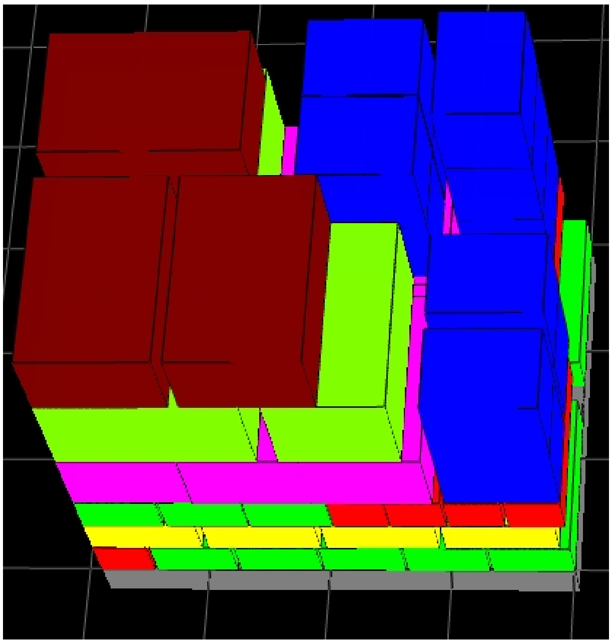

Figure 3: Display of partially constructed pallet from the pallet viewer application

The application simulates the exact execution of the final packing plan by placing one article at a time on the pallet. Metrics are computed for each article as it is added, and pallet-wide metrics are also displayed. A sample of the pallet display may be seen in Figure 3. The user is able to rotate, translate, and zoom this image as well as adding or subtracting articles from the pallet.

5. ICRA VIRTUAL MANUFACTURING COMPETITION

The first real trial of the metrics being developed for this effort occurred during the 2010 Virtual Manufacturing Automation Competition (VMAC) [3] that was part of the IEEE International Conference on Robotics and Automation (ICRA) robot challenge. During this event, three different palletizing approaches were evaluated through the use of our metrics. These approaches included a university-created neural network learning-based approach, a university-created deterministic planning approach, and a commercial product that is commonly used by industry.

At the VMAC event, teams were presented with the XML order file and were required to generate a compliant XML packlist file. This file was then passed through the Pallet Viewer software and the values of the metrics were computed. While this provided a measure of the final pallet's quality, it did not evaluate the ability to build the pallet. Therefore, teams were next tasked with running their packlist file on a simulated palletizing cell. This cell was implemented in the Unified System for Automation and Robot Simulation (USARSim) and is shown in Figure 4. USARSim performs a physics simulation that includes friction and gravity, so that problems such as sliding and tipping (which Pallet Viewer will not find) are evident. USARSim runs in real time, however, so it is not able to evaluate plans quickly.

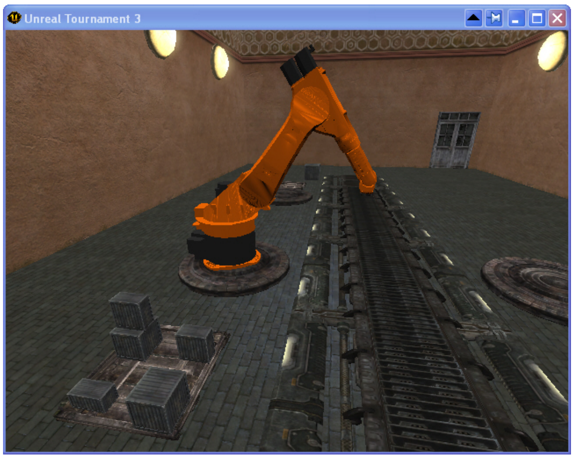

Figure 4: View of the USARSim physics-based simulation of a mixed-pallet under construction.

USARSim provided an "as-built" file at the end of the pallet's construction that could be utilized by the Pallet Viewer software for evaluation of the correctness of the build.

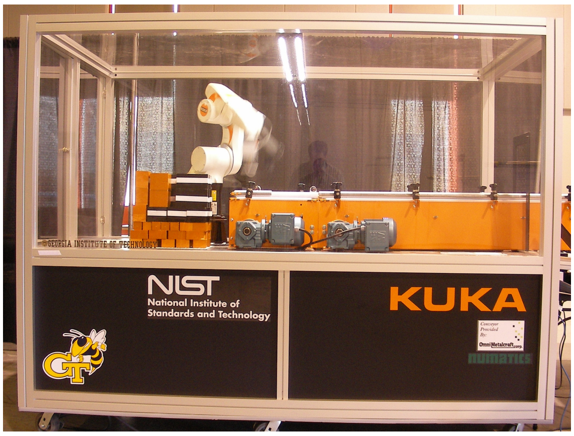

Figure 5: 1/3 scale palletizing cell used during the ICRA Competition

The final step of the competition was to allow successful teams to try and build their pallets on a 1/3 scale palletizing cell shown in Figure 5. This competition is an ongoing event and will be held during the 2011 ICRA conference. Sample palletizing code may be found through the VMAC website and new teams are encouraged to participate.

6. CONCLUSIONS

Due to the lack of widely accepted metrics, it is currently very difficult to judge the quality of a mixed pallet solution. In addition, most of the current mixed palletizing solutions are catered to a particular industry. There is a desire to obtain more flexible solutions and techniques to compare these solutions. For this, we have proposed generic problem modeling and benchmarking techniques. By comparing the mixed palletizing technique to the knapsack problem we have tried to theoretically reason benchmarks.

In our future work, we endeavor to grow as a community dedicated to the study and analysis of automation in logistics.

7. REFERENCES

[1] S. Balakirsky, T. Kramer, and F. Proctor. Metrics for mixed pallet stacking. In *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, 2010.

[2] E. Baltacioglu. The distributer's three-dimensional pallet-packing problem: A human intelligence-based heuristic approach. Master's thesis, Air Force Institute of Technology, 2001.

[3] VMAC Executive Committee. Virtual manufacturing automation competition homepage. `http://www.vma-competition.com`, June 2010.

[4] Great Little Box Company. Palletization. `http://www.greatlittlebox.com/packaging/storing`, June 2010.

[5] D. Faulkner and S. Murphy. Industrial robots in warehousing and distribution operations. In *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, 2010.

[6] P. Kolhe and H. Christensen. Planning in logistics: A survey. In *Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop*, 2010.

[7] S. Martello, D. Pisinger, and D. Vigo. The three-dimensional bin packing problem. *Operations Research*, pages 256–267, 2000.

[8] S. Martello and D. Vigo. Exact solution of the two-dimensional finite bin packing problem. *Management science*, 44(3):388–399, 1998.

[9] NIST. Pallet viewer user's manual. `http://sourceforge.net/apps/mediawiki/moast/index.php?title=Pallet_Viewer_User%27s_Manual`, November 2010.

[10] L. Sweet. Mixed palletizing in transformed supply chains: Operational requirements and technical challenges. Talk presented at Proceedings of the Performance Metrics for Intelligent Systems (PerMIS) Workshop, 2010.