item at the top of the aisle-based list; adding, by the computing system, the first candidate pick item to a bottom layer of a pallet; retrieving, by the computing system, strength information of the first candidate pick item; determining, by the computing system, an amount of weight of candidate pick items from the aisle-based list that can be layered on top of the first candidate pick item; determining, by the computing system, whether the first candidate pick item is able to support the determined amount of weight based on the determined amount of weight being less than the strength information; in response to determining that the first candidate pick item is able to support the determined amount of weight, keeping, by the computing system, the first candidate pick item in a pick sequence as the bottom layer of the pallet and removing the first candidate pick item from the aisle-based list; and in response to determining that the first candidate pick item is not able to support the determined amount of weight, removing, by the computing system, the first candidate pick item from the pick sequence as the bottom layer of the pallet and keeping the first candidate pick item on the aisle-based list.

[0010] Embodiment 2 is the method of embodiment 1, further comprising: receiving, by the computing system and from a warehouse vehicle, a request for a new task; and transmitting, by the computing system and to the warehouse vehicle, the pick sequence for each of the pallets to build, wherein the pick sequence lists candidate items to be picked in an order from a back of the warehouse to the front of the warehouse.

[0011] Embodiment 3 is the method of any one of the embodiments 1 through 2, further comprising determining, by the computing system and based on the sorted aisle-based list, a number of pallets to build per aisle.

[0012] Embodiment 4 is the method of any one of the embodiments 1 through 3, wherein the strength information includes a maximum weight load that the first candidate pick item can support without being crushed.

[0013] Embodiment 5 is the method of any one of the embodiments 1 through 4, wherein the maximum weight load is determined, by the computing system and based on: receiving a pallet of the first candidate pick item from a supplier; identifying a number of layers of the first candidate pick item on the pallet; identifying a weight for each layer on the pallet; determining a supplier-provided load on a bottom layer of the pallet, wherein the supplier-provided load is the weight for each layer multiplied by one less than the number of layers on the pallet; and

disclosed technology can provide for selecting an aisle that contains items from a pick order request. For that aisle, first items closest to the front of the warehouse can be selected first. Second items can then be selected and added beneath the first items in order to determine whether the second items are able to support weight of the first items without being crushed. Iterative decision-making can be performed to determine whether whatever weight may be borne on top of a bottom layer of the pallet is how much weight the bottom layer can support without being crushed. If the item that would go on the bottom layer cannot support the weight, then another item in the aisle can be selected to determine whether it can be borne by the bottom layer.

[0031] The disclosed technology can also be computationally efficient, thereby allowing for more pallet build sequencing decisions to be made in real-time. For example, if there are 10 different items on a pick order, there are 10 factorial (10! = 3,628,800) different possible pick and build sequences for a single pallet. Attempting to evaluate each of these pick sequences against each other, including simulation and evaluation of whether they will be structurally sound, efficient, and will avoid crushing items in the pick order, creates a significant computational burden. And this burden is only increased as the number of items on the pick order increases, and as varied groupings of items on different pallets are also considered. Accordingly, the disclosed innovation provides a method that is capable of arriving at a good solution that can balances multiple competing factors (i.e., minimize travel and pick time, avoid crushing items, build pallet that is structurally sound and unlikely to tip/lean) in a manner that is computationally efficient and that minimizes the use of computational resources to arrive at the solution. For example, the disclosed innovation can arrive at a solution by considering only a small subset of all possible combinations, and can do so without necessarily comparing solutions against each other. The disclosed technology can permit for pick solutions to be generated in real time and in a highly responsive manner, which can be beneficial in a warehousing system that is handling large volumes of similar requests and that needs to be able to provide responses with low latency to avoid backups within the warehouse.

[0032] The details of one or more implementations are set forth in the accompanying drawings and the description below. Other features and advantages will be apparent from the description and drawings, and from the claims.

pallet 116A, and determine whether the item 106B will crush item 106C, which is next closest to the front 102A of the warehouse 102 and the next nearest source item, when the item 106C is placed beneath the item 106B. In the bread example mentioned above, if item 106B is bread and item 106C is cartons of eggs, the computer system 110 can determine that the bread will not crush the eggs if the eggs are positioned beneath the eggs. Thus, the item 106B can be a layer 3 of the pallet 116A build and the item 106C can be a layer 2 of the pallet 116A build. The computer system 110 can then check whether positioning the item 106A as a bottom layer will be able to support both items 106B (layer 3) and 106C (layer 2). For example, item 106A can be boxes of cereal. The computer system 110 can determine that item 106A may not be crushed under the weight of both the bread (layer 3) and the eggs (layer 2). Accordingly, the computer system 110 can determine pick sequence 124A for pallet 116A, as shown in FIG. 1.

[0057] The pick sequence 124A lists the items to pick in reverse order from the evaluation order 118. Pick order 120 is in reverse order because the items 106A-N are to be picked from the back 102B to the front 102A of the warehouse 102. Picking the items 106A-N from the back 102B to the front 102A can be advantageous to save energy and gain on speed in building the pallets 116A-B. Moreover, it can be more efficient for the vehicle 104A to carry fewer items by traveling from the back 102B to the front 102A. Thus, where the items 106A-N are evaluated from a top-down approach (e.g., the evaluation order 118), the items 106A-B are actually picked from a back-to-front approach (e.g., the pick order 120). The pick sequence 124A lists a quantity of 20 of item 106A to be picked first because item 106A can make up the bottom layer on the pallet 116A and item 106A is at the back 102B of the warehouse 102. 40 of item 106B are to be picked second to make up the second layer of the pallet 116A. Finally, 40 of item 106B are to be picked last to make up the third layer of the pallet 116A, and item 106B is located at the front 102A of the warehouse 102. The vehicle 104A can travel along route 114A to pick the items 106A, 106C, and 106B in order.

[0058] As shown in this example, only 40 of the requested 50 of item 106B can be picked for the pallet 116A. This can occur because the computer system 110 may determine that the pallet 116A, as built using the pick sequence 124A, satisfies a pallet height threshold. The pallet height threshold can be 60 inches. Once the pallet height threshold is satisfied, additional items may not be added to the pallet 116A because then the pallet 116A would be too tall. If the pallet 116A

sequence(s) for combining the partial pallets in 440. The output can indicate a sequence to layer the partial pallets into a final pallet. The output can also indicate a set of coordinates indicating where each partial layer and/or loose item can be placed on the pallet. The coordinates can, for example, start with (0, 0) in a lower left corner of the pallet from a perspective of a warehouse worker or warehouse vehicle building the pallet.

[0105] The output can from blocks 436 and 440 can be provided to a warehouse vehicle and/or a device of a warehouse worker. As described in reference to FIG. 1, the output can be provided when the warehouse vehicle and/or the warehouse worker request to start a new task.

[0106] FIGS. 5A-B is a flowchart of a process 500 for determining an optimal pallet build using scoop and go, full, and partial layers. The process 500 can be performed by the computer system 110 described herein. One or more blocks of the process 500 can also be performed by other computer systems, servers, and/or devices. For illustrative purposes, the process 500 is described from a perspective of a computer system. FIG. 5C is a block diagram for determining the optimal pallet build of FIGS. 5A-B.

[0107] Referring to the process 500 in FIGS. 5A-C, the computer system can receive a list of candidate items for a pick order request in 502. As shown in FIG. 5C, candidate items list 522A can indicate items to be picked based on their SKU (e.g., identifier, barcode, label, QR code). The candidate items list 522A can also indicate a layer type for each of the candidate items. For example, item A is a partial layer, item B is a full layer, item C is a partial layer, item D is a full layer, and item E is a scoop and go layer.

[0108] The computer system can identify a scoop and go opportunity in 504 (refer to step A, 526, in FIG. 5C). A scoop and go opportunity can exist when a number of cases or items needed from a source pallet is greater than 50% of a total number of cases on that source pallet. Scoop and go opportunities can also exist in a variety of different scenarios, for example, when the number of cases or items needed from the source pallet is greater than 50%, 40%, 30%, 20%, 10%, 5%, etc. of a quantity remaining on the source pallet. As another example, each physical mixed pallet build can have at most one scoop and go item. However, there often can be multiple potential scoop and go items for a particular mixed pallet. In such scenarios, the computer system can identify an order line item with a highest quantity, and select that item as the scoop and go opportunity. As an illustrative example, if source pallet A has 50 cases on it and the

B and D at the top of the list 522C, since they are full layers. The items A and C have been moved to the end of the list 522C because they are partial layers.

[0114] The computer system can sort the first items based on their location from a front to a back of the warehouse in 516. The computer system can also sort the second items based on their location from the front to the back of the warehouse in 518 (refer to step C, 530, in FIG. 5C). The items can be sorted within their groupings because, as described throughout this disclosure, building a pallet from the back to the front of the warehouse can improve warehouse efficiency, reduce travel time, and reduce time needed to build the pallet.

[0115] Using the sorted list of candidate items, the computer system can determine pallet(s) and pick order sequence(s) in 520, as described throughout this disclosure (refer to FIGS. 2-4). Therefore, the computer system can first determine pallet build sequences for the full layers. The computer system can then determine pallet build sequences for the partial layers (refer to step D, 532, in FIG. 5C).

[0116] As demonstrated in FIG. 5C, the computer system can determine pallet pick order 524B. Item E can be picked first since it is the scoop and go opportunity. Item B can be picked next and layered on top of the item E since item B is a full layer and can support weight of the items D, C, and A. Item D can then be picked and layered on top of the item B since item D is a full layer and can support weight of the items C and A. Item C can be picked next and layered on top of item D because there are no remaining full layers to sequence. Of the partial layers, item C can be layered above item D and beneath item A because item C can support weight of the item A. Finally, item A can be layered on a top of the pallet since it is the last remaining partial layer and is a weight that can be supported by the items E, B, D, and C. As a reminder, items such as item E and B can also be closest to the back of the warehouse while item A can be located closest to the front of the warehouse. The pallet pick order 524B therefore can ensure that the pallet is built efficiently from the back to the front of the warehouse while using minimum travel time and energy.

[0117] FIG. 6 is a flowchart of a process 600 for determining a maximum amount of weight that a layer can support. The maximum amount of weight can be inferred based on available data, as described further below. In some implementations, the process 600 can be performed at one time for a layer of items. For example, the process 600 can be performed when the layer of

CLAIMS

WHAT IS CLAIMED IS:

1. A method for determining a pallet build sequence in an automated warehouse, the method comprising:

receiving, by a computing system, a pick order request, wherein the pick order request includes a list of items to be picked and a quantity of each of the items to be picked;

identifying, by the computing system and based on the pick order request, candidate pick items in the automated warehouse that can be used to fulfill the pick order request;

grouping, by the computing system, the identified candidate pick items based on aisles;

generating, by the computing system, aisle-based lists for each of the groups of identified candidate pick items;

sorting, by the computing system, each of the aisle-based lists based on a location of each of the identified candidate pick items in each of the groups relative to a front of the automated warehouse, wherein a first candidate pick item closest to the front of the automated warehouse is put at a top of the aisle-based list;

iteratively performing the following for each of the aisle-based lists:

selecting, by the computing system, a first candidate pick item at the top of the aisle-based list;

adding, by the computing system, the first candidate pick item to a bottom layer of a pallet;

retrieving, by the computing system, strength information of the first candidate pick item;

determining, by the computing system, an amount of weight of candidate pick items from the aisle-based list that can be layered on top of the first candidate pick item;

determining, by the computing system, whether the first candidate pick item is able to support the determined amount of weight based on the determined amount of weight being less than the strength information;

in response to determining that the first candidate pick item is able to support the determined amount of weight, keeping, by the computing system, the first candidate pick item in a pick sequence as the bottom layer of the pallet and removing the first candidate pick item from the aisle-based list; and

in response to determining that the first candidate pick item is not able to support the determined amount of weight, removing, by the computing system, the first candidate pick item from the pick sequence as the bottom layer of the pallet and keeping the first candidate pick item on the aisle-based list;

generating, by the computing system, pallet build instructions based on the pick sequence for the aisle-based list, wherein the pallet build instructions indicate that a subset of the layers of the pallet have a same build order permitting them to be assembled at interchangeable positions within the pallet when building the pallet according to the pallet build instructions; and

transmitting, to an automated layer picker, data representing the pallet build instructions that, when executed by the automated layer picker, cause the automated layer picker to automatically perform operations that include (i) moving to an aisle in the automated warehouse that has a relationship with the aisle-based list, (ii) picking items in the aisle based on the pick sequence for the aisle-based list, and (iii) assembling the picked items on the pallet based on a set of coordinates in the pallet build instructions that indicate locations on the pallet where each of the picked items should be placed,

wherein the strength information includes a maximum weight load that the first candidate pick item can support without being crushed, wherein the maximum weight load is determined, by the computing system, in a process comprising:

determining a supplier-provided load on a bottom layer of a pallet having the first candidate pick item, wherein the supplier-provided load is a weight for each layer multiplied by one less than a number of layers on the pallet; and

determining the maximum weight load based on applying a margin threshold multiplier to the supplier-provided load.

2. The method of claim 1, further comprising:

receiving, by the computing system and from a warehouse vehicle, a request for a new task; and

transmitting, by the computing system and to the warehouse vehicle, the pick sequence for each of the pallets to build, wherein the pick sequence lists candidate items to be picked in an order from a back of the automated warehouse to the front of the automated warehouse.

3. The method of claim 1, further comprising determining, by the computing system and based on the sorted aisle-based list, a number of pallets to build per aisle.

4. The method of claim 1, wherein the maximum weight load is determined, by the computing system , in a process that further comprises:

receiving data representing the pallet having the first candidate pick item from a supplier;

identifying based on processing the received data, the number of layers of the first candidate pick item on the pallet; and

identifying , based on processing the received data, the weight for each layer on the pallet.

5. The method of claim 1, wherein the margin threshold multiplier is 1.2.

6. The method of claim 1, further comprising, in response to determining that the first candidate pick item is able to support the determined amount of weight:

selecting, by the computing system, a second candidate pick item at the top of the aisle-based list;

adding, by the computing system, the second candidate pick item to the bottom layer of the pallet beneath the first candidate pick item;

retrieving, by the computing system, strength information of the second candidate pick item;

determining, by the computing system, an amount of weight of candidate pick items from the aisle-based list that can be layered on top of the first candidate pick item;

determining, by the computing system, whether the second candidate pick item is able to support a weight of the first candidate pick item;

in response to determining that the second candidate pick item is able to support the weight, keeping, by the computing system, the second candidate pick item in the pick sequence as the bottom layer of the pallet and removing the second candidate pick item from the aisle-based list; and

in response to determining that the second candidate pick item is not able to support the weight, removing, by the computing system, the second candidate pick item from the pick sequence as the bottom layer of the pallet and keeping the second candidate pick item on the aisle-based list.

7. The method of claim 1, wherein the first candidate pick item is closest to the front of the automated warehouse.

8. The method of claim 1, further comprising:

determining, by the computing system, whether the pallet having the first candidate pick item as the bottom layer exceeds a height threshold;

in response to determining that the pallet exceeds the height threshold, removing, by the computing system, the first candidate pick item from the pick sequence as the bottom layer and keeping the first candidate pick item in the aisle-based list; and

in response to determining that the pallet does not exceed the height threshold, keeping, by the computing system, the first candidate pick item in the pick sequence as the bottom layer and removing the first candidate pick item from the aisle-based list.

9. The method of claim 8, wherein the height threshold is 60 inches.

10. The method of claim 1, further comprising:

determining, by the computing system, whether pick sequences are generated for partial pallets based on the aisle-based lists; and

in response to determining that pick sequences are generated for partial pallets, generating, by the computing system, instructions to combine one or more of the partial pallets.

11. The method of claim 1, further comprising:

identifying, by the computing system and for each of the aisle-based lists, a scoop and go candidate pick item that has at least a minimum threshold quantity of the candidate pick item to fulfill the pick order request;

moving, by the computing system, the scoop and go candidate pick item to the pick sequence as the bottom layer of the pallet;

removing, by the computing system, the scoop and go candidate pick item from the aisle-based list;

identifying, by the computing system, first items in the aisle-based list that are full layers;

identifying, by the computing system, second items in the aisle-based list that are partial layers;

moving, by the computing system, the first items to the top of the aisle-based list;

moving, by the computing system, the second items to an end of the aisle-based list;

sorting, by the computing system, the first items based on location from the front to a back of the automated warehouse;

sorting, by the computing system, the second items based on location from the front to the back of the automated warehouse; and

determining, by the computing system and for each of the first items and the second items, first and second orders in which the first and second items can be picked to build the pallet having the scoop and go candidate pick item as the bottom layer.

12. The method of claim 11, wherein the minimum threshold quantity of the candidate pick item comprises a current available quantity on a source pallet for the candidate pick item having half or more of a pick quantity for the candidate pick item in the pick order request.

13. The method of claim 11, further comprising:

identifying, by the computing system, a plurality of scoop and go candidate pick items from a plurality of source pallets; and

selecting, by the computing system, one of the plurality of scoop and go candidate pick items based at least in part on the selected scoop and go candidate pick item having a greatest quantity of the scoop and go candidate pick item on a source pallet from amongst the source pallets for the plurality of scoop and go candidate pick items.

14. The method of claim 1, further comprising:

generating, by the computing system, pallet build instructions based on the pick sequence for the aisle-based list; and

transmitting, to a computing device of a warehouse worker, data representing the pallet build instructions that, when executed by the computing device, cause the computing device to output, in a graphical user interface (GUI) display at the computing device, the pallet build instructions, wherein the pallet build instructions include, for each of the aisle-based lists, (i) directions that route the warehouse worker to locations of items to be picked based on the pick sequence for the aisle-based list and (ii) sets of coordinates indicating where each picked item should be placed on the pallet to assemble the pallet based on the pick sequence.

15. A system for determining a pallet build sequence in an automated warehouse, the system comprising:

one or more processors; and

memory storing instructions, that when executed by the one or more processors, cause the one or more processors to perform operations comprising:

receiving a pick order request, wherein the pick order request includes a list of items to be picked and a quantity of each of the items to be picked;

identifying, based on the pick order request, candidate pick items in the automated warehouse that can be used to fulfill the pick order request;

grouping the identified candidate pick items based on aisles;

generating aisle-based lists for each of the groups of identified candidate pick items;

sorting each of the aisle-based lists based on a location of each of the identified candidate pick items in each of the groups relative to a front of the automated warehouse, wherein a first candidate pick item closest to the front of the automated warehouse is put at a top of the aisle-based list;

iteratively performing the following for each of the aisle-based lists:

selecting a first candidate pick item at the top of the aisle-based list;

adding the first candidate pick item to a bottom layer of a pallet;

retrieving strength information of the first candidate pick item;

determining an amount of weight of candidate pick items from the aisle-based list that can be layered on top of the first candidate pick item;

determining whether the first candidate pick item is able to support the determined amount of weight based on the determined amount of weight being less than the strength information;

in response to determining that the first candidate pick item is able to support the determined amount of weight, keeping the first candidate pick item in a pick sequence as the bottom layer of the pallet and removing the first candidate pick item from the aisle-based list; and

in response to determining that the first candidate pick item is not able to support the determined amount of weight, removing the first candidate pick item from the pick sequence as the bottom layer of the pallet and keeping the first candidate pick item on the aisle-based list;

generating pallet build instructions based on the pick sequence for the aisle-based list, wherein the pallet build instructions indicate that a subset of the layers of the pallet have a same build order permitting them to be assembled at interchangeable positions within the pallet when building the pallet according to the pallet build instructions; and

transmitting, to an automated layer picker, data representing the pallet build instructions that, when executed by the automated layer picker, cause the

automated layer picker to automatically perform operations that include (i) moving to an aisle in the automated warehouse that has a relationship with the aisle-based list, (ii) picking items in the aisle based on the pick sequence for the aisle-based list, and (iii) assembling the picked items on the pallet based on a set of coordinates in the pallet build instructions that indicate locations on the pallet where each of the picked items should be placed,

wherein the strength information includes a maximum weight load that the first candidate pick item can support without being crushed, wherein the maximum weight load is determined, by the computing system, in a process comprising:

determining a supplier-provided load on a bottom layer of a pallet having the first candidate pick item, wherein the supplier-provided load is a weight for each layer multiplied by one less than a number of layers on the pallet; and

determining the maximum weight load based on applying a margin threshold multiplier to the supplier-provided load.

16. The system of claim 15, wherein the operations further comprise:

receiving, from a warehouse vehicle, a request for a new task; and

transmitting, to the warehouse vehicle, the pick sequence for each of the pallets to build, wherein the pick sequence lists candidate items to be picked in an order from a back of the automated warehouse to the front of the automated warehouse.

17. The system of claim 15, wherein the operations further comprise determining, based on the sorted aisle-based list, a number of pallets to build per aisle.

18. The system of claim 15, wherein the operations further comprise determining the maximum weight load based on:

receiving data representing the pallet having the first candidate pick item from a supplier;

identifying , based on processing the received data, the number of layers of the first candidate pick item on the pallet; and

identifying , based on processing the received data, the weight for each layer on the pallet.

19. The system of claim 15, wherein the operations further comprise, in response to determining that the first candidate pick item is able to support the determined amount of weight:

selecting a second candidate pick item at the top of the aisle-based list;

adding the second candidate pick item to the bottom layer of the pallet beneath the first candidate pick item;

retrieving strength information of the second candidate pick item;

determining an amount of weight of candidate pick items from the aisle-based list that can be layered on top of the first candidate pick item;

determining whether the second candidate pick item is able to support a weight of the first candidate pick item;

in response to determining that the second candidate pick item is able to support the weight, keeping the second candidate pick item in the pick sequence as the bottom layer of the pallet and removing the second candidate pick item from the aisle-based list; and

in response to determining that the second candidate pick item is not able to support the weight, removing the second candidate pick item from the pick sequence as the bottom layer of the pallet and keeping the second candidate pick item on the aisle-based list.

20. The system of claim 15, wherein the operations further comprise:

generating pallet build instructions based on the pick sequence for the aisle-based list; and

transmitting, to a computing device of a warehouse worker, data representing the pallet build instructions that, when executed by the computing device, cause the computing device to output, in a graphical user interface (GUI) display at the computing device, the pallet build instructions, wherein the pallet build instructions include, for each of the aisle-based lists, (i) directions that route the warehouse worker to locations of items to be picked based on the pick

sequence for the aisle-based list and (ii) sets of coordinates indicating where each picked item should be placed on the pallet to assemble the pallet based on the pick sequence.