# Methods, Systems and Apparatus For Controlling Movement of Transporting Devices

The present invention relates to methods systems and apparatus for controlling movement of transporting devices. More specifically but not exclusively it relates to storage systems and methods for retrieving units from a storage system. In particular, but not exclusively, the invention further relates to systems and methods for coordinating and controlling product movement.

Certain commercial and industrial activities require systems that enable the storage and retrieval of a large number of different products.

One known system for the storage and retrieval of items in multiple product lines involves arranging storage bins or containers on rows of shelves arranged in aisles. Each bin or container holds one or more products of one or more product types. The aisles provide access between the rows of shelves, so that the required products can be retrieved by operatives or robots that circulate in the aisles.

It will be appreciated, however, that the need to provide aisle space to access the products means that the storage density of such systems is relatively low. In other words, the amount of space actually used for the storage of products is relatively small compared to the amount of space required for the storage system as a whole.

For example, online retail businesses selling multiple product lines, such as online grocers and supermarkets, require systems that are able to store tens or even hundreds of thousands of different product lines. The supply chains and warehouse operations of these businesses are highly dependent on their ability to organise, retrieve and return items to various containers.

In particular implementations of various warehouse and storage facility designs, containers may be stacked on top of one another and the stacks may be arranged in rows. The containers may then be accessed from above, removing the need for aisles between the rows and allowing more containers to be stored in a given volume or area.

In some of these implementations, the containers are accessed by one or more robotic or automated means, which navigate through a grid of pathways to access containers for a variety of different operations, such as moving a container from one location to another for handling, conducting operations upon a container, returning a

container to a position in a warehouse, etc.

The co-ordination of the movement of the one or more robotic or otherwise automated means may be an important consideration in determining the overall efficiency and scalability of a system for storage and retrieval of a large number of different products.

In US 6,654,662 and EP 1037828, a storage and retrieval system is described wherein "parallelepiped-shaped" containers are "deep-stacked and stock joined" in a vertical framework, forming several horizontal layers of containers whose positions at any time are random. A computer system continuously monitors and records the positions of the containers; desired containers are retrieved from the top of the framework, unwanted containers being moved, one at a time, and relocated to temporary locations until the desired container is retrieved, at which point the temporarily-relocated containers are returned to the same stack, and replaced in the same relative order; and the desired container is (eventually) returned to the top of original stack.

The system described in US 6,654,662 and EP 1037828 has several horizontal co-ordinate forming layers of containers whose positions at any time are random. Further, temporarily-relocated containers are returned to the original stack, so that their relative order is retained, and wherein the desired container is returned to the top of the stack.

In contrast to the provided system, it may be advantageous to provide deliberate container placement that may be optimised in the horizontal and the vertical domains based on different criteria such as traffic flows, frequency of access (historic, current and forecast), specific container groupings, access time, fire-resistance and or other environmental partitions.

There is need, therefore, for systems and processes for coordinating and controlling product movement.

According to some embodiments of the invention, a system is provided for controlling the movement of one or more of transporting devices transporting a plurality of objects and operating in a facility having a plurality of pathways, the system comprising:

one or more computers configured to execute computer instructions which when

executed provide:

one or more utilities determining and reserving routes for moving the one or more transporting devices through the plurality of pathways; and

one or more utilities providing a clearance system for permitting or stopping movement of the one or more transporting devices to avoid collisions.

In some embodiments, the system further comprises one or more utilities configured to optimise the movement and actions of the one or more transporting devices through the plurality of pathways.

In some embodiments, the system further comprises one or more utilities configured to optimise the placement of the plurality of objects by the one or more transporting devices in the facility.

In some embodiments, the system further comprises one or more utilities that control movement or operations conducted within one or more workstations.

In some embodiments, the one or more utilities that optimise the movement and actions utilize pathfinding algorithms.

In some embodiments, the one or more utilities optimise the movement and actions using congestion mitigation techniques.

In some embodiments, the one or more utilities optimise the movement and actions using machine learning techniques.

In some embodiments, the one or more utilities optimise the placement of the plurality of objects within the facility, and update stock levels of the facility based on the placement of one or more of the plurality of objects within the facility.

In some embodiments, the one or more utilities control the movement of the one or more of transporting devices based at least on throughput of the one or more workstations.

In some embodiments, the one or more utilities provide a clearance system configured to be tolerant to missed communications with at least one of the one or more transporting devices.

In some embodiments, the one or more utilities determine and reserve routes, and instruct a plurality or transporting devices to cooperate in transporting one or more of

the plurality of objects.

In some embodiments, the one or more utilities determine and reserve routes to move idle transporting devices from otherwise optimal routes for other transporting devices.

In some embodiments, one or more of the plurality of objects are stored within a plurality of containers.

In some embodiments, the facility stores the plurality of containers in a plurality of stacks.

In some embodiments, the one or more utilities determine and reserve routes for transporting one or more of the plurality of containers.

In some embodiments, the one or more utilities determine and reserve routes to control one or more of the plurality of transporting devices to retrieve one or more of the plurality of containers from within one or more of the plurality of stacks.

In some embodiments, retrieving the one or more containers from within the one or more stacks further requires moving one or more other containers in the stack prior to accessing the one or more containers for retrieval.

In some embodiments, moving the one or more other containers comprises placing each of the one or more other containers in an optimised position within the facility.

In some embodiments, the facility is divided into a plurality of sub-grids to reduce processing complexity.

In some embodiments, one or more control commands to control the movement of the plurality of transporting devices is provided to the plurality of transporting devices in advance of the movement of the plurality of transporting devices.

In various aspects, the disclosure herein provides methods, systems, and corresponding machine-executable coded instruction sets for coordinating and controlling product movement in at least a semi-automated order fulfillment system comprising a holding facility. In various aspects, the disclosure provides improvements in the coordination and control of the movement of robots handling a variety of goods in fulfillment of orders which, in some instances, may include a variety of items having different sizes, weights, fragilities and other characteristics.

In various embodiments of the above and other aspects, such a holding facility can include one or more storage apparatuses. In the same or other embodiments, at least a portion of the holding facility may be configured to move dynamically.

In the same or other embodiments, the holding facility may be shared between two or more workstations.

The invention will now be described with reference to the accompanying diagrammatic drawings, the drawings being to be exemplary and not limiting, and in which like references are intended to refer to like or corresponding parts.

**FIG. 1** is an illustrative diagram providing a generic computer hardware and software implementation of certain aspects, as detailed in the description.

**FIG. 2** provides a sample block diagram of the system, according to some embodiments of the invention.

**FIG. 3** provides a sample block diagram of the robot control system in more detail, according to some embodiments of the invention.

**FIG. 4** provides a sample workflow for a sample recursive path-finding algorithm, according to some embodiments of the invention.

**FIG. 5** provides a sample heat map, according to some embodiments of the invention.

**FIG. 6a** provides a table demonstrating how a search branch's projected cost could change as a result of using different coefficients in the path search algorithm. These coefficients could be derived using refinement by a machine learning technique according to some embodiments of the invention.

**FIG. 6b** illustrates how the use of different cost coefficients demonstrated in **FIG. 6a** changes which branch will be selected by the next iteration of the search algorithm.

**FIG. 7** provides a sample perspective view of a warehouse, according to some embodiments of the invention.

**FIG. 8** provides a sample diagram of a robot with a winch and a container, according to some embodiments of the invention.

Preferred embodiments of methods, systems, and apparatus suitable for use in implementing the invention are described through reference to the drawings.

Fully- and semi-automatic goods storage and retrieval systems, various aspects of which may sometimes be referred to as "order fulfillment," "storage and retrieval," and/or "order picking" systems, can be implemented in a wide variety of types and forms. One manner of providing access to goods stored for fully- and/or semi-automatic retrieval, for example, comprises placement of goods, which may be of any desired type(s), in bins or other containers (hereinafter referred to generically as containers), and stacking and/or otherwise disposing the containers in racking or vertically in layers, such that individual containers may be accessible by wholly or partially-automated container retrieval systems.

In some embodiments, the systems may include systems beyond goods storage and retrieval, such as systems where goods are processed, repaired, manipulated, assembled, sorted, etc., and the movement of goods, products, parts, components, subcomponents is required, both within a facility and/or to other facilities or transportation.

For the purposes of this specification, a storage facility for the storage, retrieval, processing and/or fulfillment of orders, wherein access to such goods is provided by fully or semi-automatic retrieval, is referred to as a "hive". The "hive" may be comprised of a grid-like layout of the potential pathways for the movement of robotic elements or devices ("robot") to traverse and perform operations at various locations in the "hive" (referred to as the "grid").

The specification is not limited to only systems that have "hives", "grids", and/or "robots", but systems that broadly control and/or coordinate the movement and/or activities of a plurality of devices may also be contemplated. These devices may be configured for the transportation of various items, such as goods and/or products, and/or containers that may be empty and/or holding such goods and/or products. These devices may further be involved in the fulfillment of orders but may also be involved in any other type of activity, such as transporting containers to and from workstations, moving objects from source locations to target locations, etc.

As indicated, the devices may be robots, and the devices may be configured to move around a hive, and/or communicate with a control system to coordinate / receive instructions on their movement. In some embodiments, the devices may be configured to communicate amongst themselves, and/or coordinate movement

amongst themselves. Accordingly, the devices may have various transporting means, communications means, powering means, processing means, processor means, sensor means, monitoring means, on-board workstations, electronic / physical storage means and/or lifting/transporting means (such as a winch, arms, etc.).

While the devices may be configured to receive instructions from the system, there may be situations where the devices lose communications with the system, have degraded communications pathways and/or do not receive communications from the system within a particular time frame.

In some embodiments, the devices may also be configured to communicate amongst each other, and/or sense the presence of each other. These communications and/or sensory inputs may be utilized, for example, in crowdsourcing information about the environment, providing redundant communications channels, verifying instructions, etc.

Fulfillment of orders may include various operations, such as, but not limited to: assembling orders where various products are purchased and aggregated for delivery to a customer, such as for a grocery chain; assembling products with various subcomponents; conducting various operations on products (such as soldering components together), sorting products, etc.

Orders may also be returned, for example, if an order is cancelled, a delivery fails, etc. In some scenarios, while an order is in the process of fulfillment within the hive, it may be cancelled and the product items may need to be returned. In some scenarios, the items may need to be placed again into containers, and the containers moved to various locations. In some scenarios, a workstation may need to conduct tasks to reject/rework products when an order is returned or cancelled.

Furthermore, as mentioned above, individual containers may be in vertical layers, and their locations in the "hive" may be indicated using co-ordinates in three dimensions to represent the robot or a container's position and a container depth (e.g. container at (X, Y, Z), depth W). In some embodiments, locations in the "hive" may be indicated in two dimensions to represent the robot or a container's position and a container depth (e.g. container at (X, Y), depth Z).

The "hive" itself may be a "dynamic" environment, in the sense that robots and

workstation locations may be associated with different parts of the hive for engaging in actions. For example, robots may need to access a specific container in a specific location in the hive dimensions (e.g. container at (X, Y, Z), depth W) to fulfil a particular order or to store a product in the "hive". This involves movements of the robots along various possible paths, for example, along the top of the grid, and then accessing particular containers at selected depths of a stack.

The access of particular containers at selected depths of a stack may necessitate the movement of containers which may otherwise obstruct the ability to access a particular container (e.g. where the containers are stacked, a number of containers must be moved first to be able to access a container that is not at an accessible end of the stack). In some embodiments, it may be advantageous to have the system configured to provide for the evaluation and optimisation of a new position for every container that has to be removed to access a target container.

Containers moved off of a stack need not be moved back to their original stack.

One of the potential advantages is the ability to modify the distribution of containers such that the containers are located in more easily accessible or otherwise more convenient locations.

This may help maintain an optimal distribution of containers within the facility, for example, biasing containers that are expected to be in higher demand in more easily accessible locations, such as locations nearby or within workstations, to reduce travel distance.

**FIG. 7** provides a sample perspective view of a warehouse, according to some embodiments of the invention.

Robots may have various shapes, sizes and configurations, and may have various communications means, sensors and tools. In some embodiments, each robot may be able to communicate with the control system through a set of frequency channels established through a set of base stations and base station controllers. Robots may utilize various tools to move and obtain containers from a stack, including, for example, a winch to carry a container.

**FIG. 8** provides a sample diagram of a robot with a winch and a container, according to some embodiments of the invention.

The grid is not limited to rectangular grid elements and may be comprised of curved tracks, tracks up and down, etc. The grid pathways may have intersections and may be accessed by more than one robot.

Each grid may be segmented, physically or logically, into one or more sub-grids.

The grid may be comprised of one or more workstations. Workstations may be manual, semi-automated or fully automated, and may consist of locations or areas where operations are conducted within the hive, or operations are conducted in relation to the hive, containers or products, such as, moving products in or out of the hive, manufacturing products, assembling products, processing products to their components, providing staging locations to support other steps or operations, etc.

Workstations could include, for example, locations where items are moved from inbound carriers, locations where products have various operations conducted on them (e.g. assembly of components, painting, sorting, packaging, disassembly, reworking products, fixing packaging, replacing products in cancelled orders, rejecting returned products, disposing products), products are moved to outbound carriers, locations with capabilities for refrigeration, locations where components or objects are assembled, locations used for staging or pre-fetching products, locations where robots are repaired and maintained, locations where robots are charged, locations where workers "pick" products to be placed into containers, locations where workers "pick" products to be removed from containers in fulfillment of orders, bags are placed into containers, etc.

Where items / products are returned to the hive, the system may support and/or control the process of bringing back the product, reworking the product, and/or disposing the product if rejected. The scenario may, in some embodiments, involve processing the returned container (which may be a delivery tote or other object as well) at a workstation to determine whether it can be accepted back into the system, whether it needs reworking / repackaging, and/or whether the product should be disposed of instead (e.g. a perishable product has expired).

Workstations may have one or more workers or robots present to conduct various tasks, such as picking items for fulfillment of orders.

In some embodiments, workstations may also be stations with conveyors, refrigerators, various tooling technologies and/or other technology to manipulate,

paint, fasten, repair, freeze, heat, expose to chemicals, refrigerate, filter, assemble, disassemble, sort, package, scan, test, transport, store or process goods, containers, etc.

The workstations may have their own pathways within the facility, share pathways with the facility, etc. The workstations may also have various input and output pathways or other types of entry / egress points within the facility.

In some embodiments, the workstations communicate with one or more warehouse management systems to provide information and data related to the status of the workstation, workflow, required containers, issues, status of products held or otherwise manipulated (e.g. sub-components being assembled together), etc.

Upon receipt of an order from a customer for multiple items stored in a storage and retrieval system, fully or semi-automated container handlers may retrieve storage containers containing relevant items from a grid, racking, or other ordered arrangement of storage containers, and deliver them to one or more workstations.

At the workstations, items may be removed from the storage containers and placed in an intermediate holding facility before being picked into delivery containers.

As indicated throughout this specification, it may be advantageous to have a distribution of containers such that containers that are likely inputs for certain workstations are located close in proximity to those workstations.

The picking of items from containers may be done manually by human pickers, or be conducted in a semi-automated or fully-automated manner with the assistance or participation by various mechanical or robotic elements.

Where individual containers are stacked vertically in layers, accessing a container that is not at the top layer may require a further set of operations to move containers stored at layers above the desired container prior to being able to access the desired container. For example, if a desired container is one level below the top layer, the container residing within the top layer may need to be moved to another location prior to accessing the desired container. As indicated throughout this specification, it may be advantageous to have a distribution of containers such that containers holding items in greater demand are biased towards the more easily accessible positions (e.g. if a vertical stack of containers, the uppermost levels). In some embodiments, an optimisation module provides an optimal location for these

containers to be positioned.

In a typical commodities picking operation adapted for handling a large variety of items, such as a grocery order processing system, it is sometimes found that a wide range of items of various sizes, shapes, weights, and other characteristics must be handled or otherwise accommodated, and that these items may need to be moved around a facility to various stations for various operations to be conducted, during the fulfillment of one or more order(s). Depending on the size, organisation and arrangement of the facility, the movement of these items may be advantageously optimised so that items are moved efficiently, collisions are avoided and dependencies are resolved (e.g. objects are picked and unloaded in a proper order).

The various actions involved in these working groups such as placement of items in different types of containers, storage of containers within the hive, and sorting/delivery as described can result in various movements of the containers, including within the hive, but also placement of containers in different locations. These actions may include inbound/outbound activity to the hive (e.g. bringing items to and from a warehouse), and may also include associated sorting/delivery systems such as management of conveying product off vehicles for storing/delivery, and also conveying onto vehicles for order fulfilment.

In other embodiments of the invention, the workstations may be utilized to provide other types of item handling or container handling, such as workstations where actions are performed on items (e.g. assembly, disassembly, modification, sorting, drying, freezing, testing, chemical exposure, physical manipulation, fastening, repair, printing, painting, cutting, packaging, storage, processing, welding, tempering, reprocessing). These workstations may be manual, semi-automated, or automated and have various parameters associated with their operation or performance, such as throughput, required inputs, required outputs, load balancing, required delays (e.g. drying time), etc.

The various movements of the robots, placement of containers / objects, and control over selecting when to remove product from containers may be controlled and optimised by one or more control systems. This control may include the implementation of various strategies, including, for example:

The central management of one or more robots;

Managing the robots not only to retrieve containers for processing, but also to "pre-fetch" containers to more convenient locations, for example, locations close to or within workstations;

Path optimisation through the application of one or more path finding algorithms (e.g. branch and bound techniques);

Path optimisation through the application of one or more heuristic techniques to the one or more path finding algorithms (e.g. projected lowest cost, where in one embodiment of the invention, cost may be calculated as a function of total time to target, and total "accumulated heat" for congestion mitigation on the grid when viewed as a whole);

Scheduling / pre-processing movement pathways in advance;

Conducting simulations to determine optimal depth of analysis in scheduling / pre-processing movement pathways;

Application of machine learning techniques to optimise exponents and coefficients applied to the cost functions of one or more algorithms. In some embodiments, this may include path finding, container placement and workstation load-balancing algorithms;

A just-in-time robot path conflict resolution system;

The subdivision of the robot population into separate control groups each containing one or more robots;

Container location optimisation through various means, such as the application of a scored selection algorithm using weighted cost coefficients;

Optimisation of the allocation of work to workstations;

Pre-processing of tasks and actions to schedule future tasks and actions.

In an example environment where one or more robots are used for the fully or semi-automated retrieval, storage and movement of objects, the warehouse may first have to decide how to distribute the container loads between different robots or groups of robots.

For example, the allocation of tasks across various groups of robots and pick stations may be optimised depending on the particular layout of a warehouse, the

placement of items, particular characteristics of goods (e.g. the item is expiring or is potentially dangerous) and the workflow ordering.

For example, it may be desirable to have a system that intelligently compares potential paths for a robot to take to its destination, taking into consideration, among others, the potential congestion along that path, the time required to complete operations, the potential for collisions, the objects held in the inventory of a particular robot, predicted future operations and the characteristics of a particular robot (e.g. battery levels, service issues).

It may further be desirable to have a system that intelligently adapts to various conditions in the "hive", such as idle robots that may hinder or block the path of a robot, obstacles, or other robots reserving paths that a particular robot is seeking to traverse.

It may further be desirable to have a system that intelligently positions containers based on an algorithm to bias the system towards a distribution of containers that aids in the efficient retrieval of items (e.g. containers with items of high-demand SKUs are kept near the top and near their workstations for ease of access).

These are non-limiting examples, and any optimisation methods, arrangements or considerations may be implemented.

Referring to **FIG. 2**, a schematic diagram is provided of sample fully- and semi-automatic goods storage and retrieval systems, according to some embodiments of the invention. **FIG. 2** is provided at a high level, illustrating an example implementation. Various implementations of the system **200** may involve more or less components and **FIG. 2** is merely provided as an example.

The system **200** is comprised of a robot control system **202**; a maintenance/monitoring system **204**; a base station controller **206**; one or more base stations **208a** and **208b**; one or more robots **210a**, **210b** and **210c**, and one or more charger stations **230**. While only two base stations **208a** and **208b**, and three robots **210a**, **210b** and **210c** are illustrated, it should be understood that that there may be more, or less robots and base stations in other embodiments of the system.

There may be one or more warehouse management systems (WMS) **232**, order management systems **234** and one or more information management systems **236**.

The warehouse management systems **232** may contain information such as items required for an order, SKU#s in the warehouse, expected / predicted orders, items missing on orders, when an order is to be loaded on a transporter, expiry dates on items, what items are in which container, whether items are fragile or big and bulky, etc.

In some embodiments, the warehouse management systems **232** may be in communication with the workstations and may also contain information related to the operation of the workstations, such as the status of the workstation, what products and/or what containers the workstation is required to receive at particular times, what products and/or containers the workstation will be required to have moved to another location at particular times, the expected workflow for operations at the workstation, the number of robots currently waiting to bring containers to a workstation, etc.

The robot control system **202** may be configured to control the navigation/routing of robots, including, but not limited to, moving from one location to another, collision avoidance, optimisation of movement paths, control of activities to be performed, etc. The robot control system **202** may be implemented using one or more servers, each containing one or more processors configured based upon instructions stored upon one or more non-transitory computer-readable storage media. The robot control system **202** may be configured to send control messages to one or more robots, receive one or more updates from one or more robots, and communicate with one or more robots using a real or near-real time protocol. The robot control system **202** may receive information indicating robot location and availability from one or more base stations **208a** and **208b**. The robot control system **202** may be configured to keep track of the number of robots available, the status of one or more robots, the location of one or more robots and/or their current instruction sets. The robot control system **202** may also be configured to process and/or send control messages to the one or more robots in anticipation of future movements, potentially reducing the processor load, and also proactively managing the traffic load with respect to the communications links. Such an implementation could be advantageous in light of complex algorithms in use by the robot control system **202** in determining optimal pathways, calculating optimal locations for containers and/or determining reservations and/or clearances.

In some embodiments, the servers may utilize a 'cloud computing' type platform for

distributed computing. A cloud-based implementation may provide one or more advantages including: openness, flexibility, and extendibility; manageable centrally; reliability; scalability; being optimized for computing resources; having an ability to aggregate information across a number of users; and ability to connect across a number of users and find matching sub-groups of interest. While embodiments and implementations of the present invention may be discussed in particular non-limiting examples with respect to use of the cloud to implement aspects of the system platform, a local server, a single remote server, a software as a service platform, or any other computing device may be used instead of the cloud.

In some embodiments, the movement optimisation module may utilize one or more control groups to segregate robots into the one or more groups. The use of control groups for large grids may provide certain advantages, such as the ability to maintain operation of a very large grid whenever real-time computation cannot keep up with re-planning after a control anomaly such as when (i) wireless communication link drops more sequential packets than allowed for in planning; (ii) one or more robots fail; (iii) one or more robots operate outside the pre-determined tolerance on performance.

A control stop message may be broadcasted to the robots in a particular "control group"; potential benefits from broadcasting messages as opposed to sending individual messages may include improved communications through the use of multiple transmission slots and a potentially higher signal to noise ratio.

In some embodiments, the robot control system **202** may be configured to dynamically assign robots to different "control areas" as they move across the grid.

The maintenance / monitoring system (MMS) **204** may be configured to provide monitoring functions, including receiving alerts from one or more robots or one or more base stations, establishing connections to query robots. The MMS **204** may also provide an interface for the configuration of monitoring functions. The MMS **204** may interact with the robot control system **202** to indicate when certain robots should be recalled.

The base station controller **206** may store master routing information to map robots, base stations, and grids. In some embodiments, there may be one base station controller **206** per warehouse, but in other embodiments of the invention, there may

be a plurality of base station controllers. The base station controller **206** may be designed to provide high availability. The base station controller may be configured to manage dynamic frequency selection and frequency allocation of the one or more base stations **208a** and **208b**.

The base stations **208a** and **208b** may be organised as a pool of base stations, which may then be configured to be active, on standby or to monitor the system. Messages may be routed through a variety of communications means to/from robots. The communications means may be any communications means, in some embodiments, the communications means may be a radio frequency link such as those falling under wireless standard 802.11. The base stations **208a** and **208b** may further include processing units **212a**, **212b**, digital signal processors **214a**, **214b**, and radios **216a**, **216b**.

The one or more robots **210a**, **210b**, and **210c** may be configured to move around the grid and to perform operations. Operations may include moving a container from one stack to another, going to a charging station to recharge, etc. The one or more robots may be assigned to communicate with the one or more base stations **208a** and **208b**.

The one or more robots **210a**, **210b**, and **210c** may not all be the same type of robot. There may be different robots with various shapes, designs and purposes, for example, there may be a robot with a footprint of a single grid position which winches containers for internal stowage, a cantilever robot, a bridge robot, a recovery robot, etc.

In some embodiments, the one or more robots **210a**, **210b** and **210c** have winches on them which may be used to retain a container for movement from one position on the grid to another.

The robots **210a**, **210b** and **210c** may have, respectively, radios **218a**, **218b**, **218c**, digital signal processors **220a**, **220b**, **220c**, processors **222a**, **222b**, **222c**, real time controllers **224a**, **224b**, **224c**, batteries **226a**, **226b**, **226c** and motors, sensors, connectors, etc., **228a**, **228b**, **228c**.

The one or more charger stations **230** may be configured to provide power to charge batteries on the one or more robots **210a**, **210b** and **210c**. The one or more charger stations **230** may further be configured to provide high speed, wired data access to

the one or more robots, and several charge stations may be placed around the grid for use by the one or more robots **210a**, **210b** and **210c**.

Referring to **FIG. 3**, a block diagram is provided of the control system **202**, according to some embodiments of the invention. The block diagram is provided for illustrative purposes to identify some of the components of control system **202** in more detail, however, not every module or interface identified may be required and, in various embodiments, more or fewer modules may be included.

The control system **202** may be configured to evaluate how to improve work allocations, movements of product and placement of product. According to various embodiments of the invention, optimisations may be run in real time, while others are run, for example, periodically during down time or less active times.

The control system **202** may be configured to schedule when specific types of movements should happen and in what order they should occur, depending on the application of various business rules, indicating priority, etc. The control system **202** is configured to determine both inbound and outbound factors in making decisions even relative to product placement for example. For example, the control system **202** may act on estimated delivery location of product supply, and estimated outbound delivery of product. The control system may make decisions, and may send signals for execution by an automatic system, and / or may allocate tasks efficiently to humans (pickers, loaders etc.).

The control system **202** may determine that one or more robots or one or more pickers should conduct one or more actions in the fulfillment of an order or for any other purpose. The action of the one or more robots may require the robots to traverse the grid, and / or to conduct actions, such as retrieving a container.

The control system **202** may be configured to analyze various pathways in the grid to determine one or more paths that may potentially be preferential relative to other pathways, given a set of constraints and conditions. These preferential pathways may then be provided, one-time, periodically and / or dynamically to the robots to control their movements throughout the grid.

A path may be preferential for a number of reasons, including, but not limited to: less distance travelled, greater expected average velocity of robot, lower probability of encountering traffic (i.e. congestion), less total time required, lower probability of

collision, less power used, ease of switching to alternate pathways, ability to avoid obstacles (e.g. a broken robot, a dropped item, a broken path, a part of the path is under repair).

The control system **202** may use various algorithms to identify, design and/or control the movement of various robots it is connected to. In some embodiments, the control system is implemented using one or more servers, each containing one or more processors configured to perform one or more sets of instructions stored upon one or more non-transitory computer readable media. Potential advantages for computer implementation include, but are not limited to, scalability, ability to handle large amounts of processing and computational complexity, increased reaction speed, ability to make decisions quickly, ability to conduct complex statistical analysis, ability to conduct machine learning, among others.

These algorithms are discussed more at depth further in the specification, and sample path-finding algorithms and heuristic approaches are provided, according to some embodiments of the invention.

Constraints may include the current layout of the grid, the physics of the robot (e.g. maximum velocity, turning radius, turning speed, maximum acceleration, maximum deceleration), congestion (e.g. expected traffic load at a certain pathway or intersection), established 'highways', impact of objects being carried by the robot (e.g. big, bulky, or fragile objects), robot status and condition (including battery condition, damage, maintenance issues), and station status (e.g. the destination station is full or temporarily blocked).

The control system **202** may be a real or near-real time control system (controlling the actions of the various units including robots and optionally the associated other units involved such as conveyors, pickers, humans, etc.). The control system **202** may be comprised of one or more modules. The one or more modules may include a control interface **302**, a movement optimisation module **304**, a product placement optimisation module **306**, a robot physics model module **308**, a business rules module **310**, a clearance module **312**, a reservation module **314**, a command generation and scheduler module **316**, a robot communications module **318**, a charge manager module **320** and an alert/notification module **322**. These modules may be implemented in various ways, in some embodiments they are implemented

as applications stored as instructions on one or more computer-readable media to be performed by one or more processors.

The control system **202** may provide real or near-real time control of the allocation of work, workstation operations, the movement of robots and/or the placement of containers, according to some embodiments of the invention. The allocation of work, movement and placement of containers may be precipitated by actions as relevant to activities within a warehouse, such as the fulfillment of orders, the redistribution of containers to more easily accessible positions, estimated dispatch sequences, maintenance operations, workstation operations, anticipation of future orders, etc.

The control interface **302** provides an interface for various external systems to provide directions and information into the control system **202**. The control interface **302** may, in various embodiments, provide interfaces for human users and/or interfaces for interfacing with various machines and systems.

Interfaces for humans may include, for example, keyboards, visual displays, command prompts, etc.

Interfaces for machines and systems may include application programmable interfaces (APIs), implemented using different specifications, including, but not limited to, simple access object protocol (SOAP) and representational state transfer (REST) services, and/or interfaces written in various programming languages. The control interface **302** may interact with various external databases, including but not limited to various warehouse management systems and order management systems; and also may receive information from the various robots (e.g. a robot is malfunctioning, a robot requires charging, a robot is en route to the destination, a robot has encountered an unexpected obstacle, etc.).

The control interface **302** may also receive and transmit information to and from the warehouse management system (WMS) relevant to the control and movement of robots and containers. Such information may include, but is not limited to, grid location and sizing, the establishment of sub-grids, master records of inventory and orders, locations of workstations, parameters related to workstations, and/or also the dispatch sequence (e.g. when orders need to go out). As actions are performed, containers brought to workstations, workstation operations completed, delivery totes filled, etc., the control interface **302** may provide updates to the WMS. In some

embodiments, there is a confirmation process between the WMS and the control interface **302**. These updates to the WMS may include, for example, updated stock levels related to particular SKU#s, updated container positions, updated object positions within containers, updated facility conditions, etc.

In some embodiments of the system, in addition to the WMS, there may also be a separate order system that contains and provides information regarding various orders entering the system, the fulfillment of orders, workstation operations, upcoming orders and predicted orders.

The control interface **302** may also receive commands to stop the operation of a particular robot, a group of robots or all of the robots (e.g. in the event of a malfunction, an emergency, etc.).

The movement optimisation module **304** may be configured to optimise the movement of robots through applying various algorithms to determine potentially advantageous routes from one location to another. The potential advantages may include shorter distance travelled, lower likelihood of encountering congestion, shorter time required, lower power consumption, co-ordination with movements of other robots, routing around obstacles such as broken robots or broken areas of track, co-ordination with various workstation operations, etc.

The movement optimisation module **304** may be configured to provide work allocation, planning and scheduling functions, including developing a set of tasks and then selecting which pick station or robot should conduct which task. For example, this may be based upon where a robot or a pick station is located, the particular capabilities of the robot or pick station, etc. Further, the particular permutations and set of actions required to fulfill a particular order are determined and actions/tasks are developed for one or more robots and or one or more pick stations. Functions may include, among others, delivering empty containers to inbound stations, placing containers loaded with goods around the warehouse, bringing containers to pick stations or other areas, moving containers from one location in the warehouse to another, etc.

The movement optimisation module **304** may be configured to interact with the product placement optimisation module **306** in determining a set of potentially advantageous locations to place an object. For example, given that a container

containing items of a particular SKU# that is required at a high frequency, the product placement optimisation module **306** may indicate that it should be placed at a certain location in a certain stack that is more accessible for retrieval. Conversely, if a container contains items of a particular SKU# that is required at a low frequency, the product may be determined to be placed at a lower depth within a less easily accessible grid location.

In optimizing movement, the movement optimisation module **304** may be configured to consider various factors involved in both movement and the performance of an operation, such as the expected time required to get to a particular location, how deep the container is within a stack, how long it would take to dig a container out of a stack, the various operations necessary to move containers located above to other locations, etc.

The movement optimisation module **304** may also be provided a set of inputs from the robot physics model module **308**, which may communicate a set of constraints on the movement of the robot depending on various factors (e.g. the robot may only move at 50% of the maximum velocity as the robot is currently carrying delicate objects). The movement optimisation module **304** may coordinate the movement of boxes into the grid, out of the grid and within the grid.

In some embodiments, the movement optimisation module **304** may dynamically recalculate preferential paths during the course of a robot's journey to potentially determine an updated set of paths as conditions and constraints change over time.

In some embodiments, the grid may be pre-processed by the movement optimisation module **304** to potentially increase processing speed and/or reduce processing load. Other methods of reducing processing load are also contemplated, such as reducing the depth/breadth of searching, dividing the grid into sub-grids, distributed processing, caching future routes, computationally simplifying the grid (e.g. reducing the number of nodes under analysis), reducing path re-calculation, etc.

In some embodiments, the movement optimisation module **304** may divide a grid into a plurality of smaller sub-grids for analysis. Such division may ease demands on processing power, which may be particularly useful if grid sizes are very large; for example, a 1000x1000 grid may be broken into 100 100x100 grids and each grid analyzed separately. This may be further useful if the system is attempting to control

a very large number of robots or take into consideration a large number of conditions.

The movement optimisation module **304** may also interact with the clearance module **312** and the reservation module **314** in determining whether the navigation of a proposed pathway will encounter issues involving the clearances and reservations of other robots and also determining pathways that may reduce the chances of encountering these issues.

Where a desired container is located within a stack at various depths within the stack, the movement optimisation module **304** may be required to control one or more robots in the movement of containers off the stack so that the desired container is accessible. The movement optimisation module 304 may coordinate movement across one or more robots such that the one or more robots cooperate in moving containers off the stack.

In some embodiments, the movement optimisation module 304, it may not be necessary or even desired to replace the containers on the stack, rather, containers that have been moved off of the stack may have an optimal position determined by the product placement optimisation module **306**, and may be moved there by the one or more robots. A potential advantage of such an embodiment is that increased efficiency may be found when containers are not replaced in their original position but rather placed in a more optimal position.

In some embodiments, the clearance module **312**, the reservation module **314** and the movement optimisation module **304** are utilized together as a path conflict resolver, wherein a movement optimisation module **304** develops a path and then reserves the path using the reservation module **314**, and the clearance module **312** provides a just-in-time approach to determining priority when robots are engaged in potentially conflicting paths.

In some embodiments, the movement optimisation module **304** is configured to account for situations where a robot is attempting to take a container to a full station. In this situation, the movement optimisation module **304** is configured to instruct the robot to take the container nearby to be held by the robot until the station can accept the container. When the station can accept the container, it is dropped off. In these embodiments, held containers may be dropped off in priority order.

In some embodiments, if a station becomes free for drop off before the held container arrives at its holding location, the movement optimisation module **304** will re-plan to drop off the container directly, without holding.

In some embodiments, the movement optimisation module **304** is further configured for pre-fetching operations, wherein a container is moved closer to a station prior to it being required at the station. Containers are then ready to be dropped off when required, which may reduce the uncertainty of the drop off time.

In some embodiments, the system may be configured to plan the robot paths and establish robot path reservations sufficiently far in the future to allow the algorithms to complete.

In some embodiments, the degree of forward planning required may be computed by simulation.

The simulations may be used to adjudicate (in a statistical sense) between the efficiency gains of planning far into the future; against the efficiency loss of having higher probabilities of having to re-plan when robots fail to maintain their plan, because of various reasons, such as short-term communication packet losses, and/or robots running outside the allowed tolerances.

The product placement optimisation module **306** may be configured to determine a set of potentially advantageous locations to place a particular container containing particular items. The product placement optimisation module **306** may utilize relevant information, such as the layout of the grid, the frequency at which a particular item is requested, future ordering, predicted future ordering, the location of workstations, the location of charging stations, and the congestion level of particular areas and branches of the path, among others, in determining the set of potentially advantageous locations to place a particular container containing particular items.

A robot may be tasked with transporting one or more containers, each of which may contain one of more items, to satisfy the demands of one or more service pick stations. Each container may be provided with an index number related to a stack location so that particular containers may be biased towards particular stack locations.

The distribution of and positioning of a container within a stack or relative to the layout of a grid may contribute towards the overall operational efficiency of a

warehouse.

There are various situations where a container needs to be placed somewhere in the facility and these situations present an opportunity to re-evaluate and/or optimise the distribution of the position of containers in the facility.

There are various considerations that would indicate that a position is better or worse than another position, which may include, but are not limited to, distance to workstations, level of congestion in the area, which containers will be blocked by a particular container, logical groupings of containers relative to environmental factors (e.g. containers containing flammable objects may require special positioning), intelligent pre-fetching to a location by a work station.

These considerations may be utilized by the system to compare one position with another, for example, by using a weighted algorithm or any other suitable method.

For example, when a new container is introduced into the facility, when a container is returned to the facility, when one or more containers are moved by robots attempting to access the items stored deep within a stack, when items are placed in/removed from the container, when a container is marked damaged, when a container is marked as dirty, etc.

This optimisation can be conducted at various times, for example, in some embodiments, the optimisation is conducted to determine a new position for every container that has to be moved, such as those that have to be removed so that a particular container below can be accessed.

For example, the number of robot movements and operations may be potentially reduced if frequently ordered items and only a certain number of items for each given SKU# are distributed in the easiest to access areas of the warehouse (e.g. the closest locations to each pick station and/or the tops of stacks of containers).

An "active window" is defined as the number of orders' worth of containers to keep in the active state (on hand to service workstations) and determined by the system. The product placement optimisation module **306** may be configured to assign scores to containers, biasing the overall hive layout to tend towards a top layer of active containers, with reserve stock underneath.

This scoring system may be determined by the business rules module **310**, and set

based upon information such as the stock expiry date. Historical orders may be utilized, among other information, to factor into calculating a score for containers, and this score may be updated on a continual basis every time the stock level of a SKU# changes. This scoring system may be utilized to help bias the positions of containers to maintain the optimality of positioning of containers within the facility.

In some embodiments, the system may be configured to control robot movement to maintain an "active" pool of one or more containers per SKU# to satisfy the demands of one or more service workstations. As an "active" pool becomes depleted, "reserve" containers may be promoted into the "active" pool.

In some embodiments of the system, the system may be configured to control robot movement using a "put-away algorithm" to determine a "best match" stack location from an available pool of empty locations when placing a container.

In some embodiments, the "active" pool may be configured to support parallel demand from the one or more service workstations.

In some embodiments, the product placement optimisation module **306** may also balance orders between the pick stations.

The robot physics model module **308** may be configured to store a set of variables that are designed to model the particular physical properties relevant to a robot. For example, the model may indicate physical characteristics such as the length, weight, height and width of a robot, the maximum carrying capacity of a robot, the rotational speed of the robot, the winch cycle time of a robot, the maximum velocity and acceleration of a robot, the ability for a robot to perform certain actions given a set amount of battery life, etc. The robot physics module **308** may interface with the business rules module **310** in determining limits on certain characteristics of robot movement, including the maximum velocity, maximum acceleration, and maximum rotational speed of a robot. For example, a robot carrying a number of cartons of eggs may be required to only accelerate/decelerate at 25% of the maximum acceleration/deceleration of the robot due to the vulnerability and fragility of the eggs due to physical forces.

The business rules module **310** develops and applies a set of business rules based upon the particular circumstances of the warehouse, robots and communications systems. For example, the business rules module **310** may provide that for certain

classes of items, various restrictions are in force for the robot physics model module **308** to potentially reduce the amount of damage incurred by goods in transit. Examples of where business rules may be implemented include high risk products (e.g. acid, bleach etc.), containers with aerosols, and containers with flammable contents, among others. Empty containers may also be treated differently to other containers.

The business rules may include actions such as cleaning a container prior to re-use, slowing down robots containing certain objects, etc.

The business rules module **310** may also be configured to develop and apply sets of rules governing the placement of products. For example, different rules may be in place for high-frequency items, items that may be picked soon due to incoming orders, etc.

The clearance module **312** may be configured to store and provide clearances for various robots. A system of clearances may be accessed to determine whether a path is clear for a robot to traverse. The clearance module **312** may be implemented as a passive collision avoidance system, wherein robots are only given the smallest amount of work possible without impacting performance.

Upon providing a robot with a new instruction, the clearance module **312** checks that it will not be possible to collide with another robot, based upon, for example, grid dimensions, grid positions, move commands generated by planning, cancellation of move commands (generated on events such as a controlled stop), the current positions and speeds of robots, braking ability of robots as well as where they have been cleared to visit.

The clearance module **312** may be configured to issue clearance "just in time", and may be used to grant permissions to robots to continue along their planned paths. A new clearance may be generated (or withheld) in response to each robot status report. As such, the clearance module **312** may act as a path conflict resolver. Where clearances are required, the clearance module **312** may interact with the movement optimisation module **304** to dynamically re-plan routes to resolve or avoid conflicts.

The clearance module **312** may provide to the control interface **302** what the clearances for a path would be, notification of when a clearance is issued (e.g. to the

planning system as this may allow dynamic re-planning from the end of the current clearance), notification of when a clearance is withheld (e.g. to identify error cases, and to identify needs to re-plan), and to an alerting system (because there is a potential problem with a robot, robot communications, or the control system **202**).

The clearance module **312** may be configured to devise clearance schemes based upon a set of tolerances, including missed messages, processing time, clock sync and robot discrepancies with the physics model, among others.

The clearance module **312** may provide a set of safe entry times for one or many position on the grid, based upon robot position and speed updates, and clearances given/withheld. The set of safe entry times may be dynamically updated as the conditions of the grid change.

The clearance module **312**, in some embodiments, may be configured such that robots are only provided clearances for a predetermined period of time (e.g. 3 seconds). The clearance given to a robot may be configured such that the period of time is sufficient for the robot to come to rest without risking a collision.

In some embodiments, the clearance module **312** may be configured such that clearances are provided such that the control system **202** may be able to miss a configurable number of status messages from the robot, and still have the robot continue operation for a short period of time. This design may result in a system that may be more tolerant to missed packets, which may be advantageous in having continued operation even where some issues with communications are encountered. The configurable number may be set such that there is a high probability that the control system **202** is likely to receive a status message from the robot before the robot autonomously decelerates to be at rest at the end of its clearance.

In some embodiments, if a robot has begun to decelerate, it will be allowed to come to rest and the control system **202** may then cancel its onward reservations and re-plans its path over the grid.

The reservation module **314** reserves various paths on the grid (e.g. robot A is planning to take path X and reserves path X during the expected traversal time. Robot B, knowing that robot A has reserved path X, chooses path Y instead). The reservation module **314** may be designed to create non-conflicting robot movement plans, and may be configured to work in conjunction with the clearance module **312**

and the movement optimisation module **304**.

The reservation module **314** may be configured to provide reservations for a robot for a grid location for a span of time, where no two robots may be given overlapping reservations, taking into account tolerances for robots being marginally off plan, tolerances for lost robot communication messages, and tolerances for clock discrepancies, among others.

In some embodiments, the reservation module **314** is used to reserve routes in advance and to make sure that robots do not plan to take conflicting paths, especially where there are a large number of robot actions and tasks taking place simultaneously. The reservation module **314** may be configured to allow sufficient tolerance for any robot to stop under controlled braking without risking a collision.

The reservation module **314** may be configured to interact with the movement optimisation module **304** to establish the robot path reservations sufficiently far in the future to enable forward planning. In some embodiments, the reservation module **314** and the movement optimisation module 304 allow sufficient forward planning to complete the computation of the movement algorithms.

The command generation and scheduler module **316** generates a set of instructions to be transmitted to the one or more robots. These instructions may include, for example, that robot A is required to move to location B to obtain container C, bring container C to a workstation and then return container C to a particular location D. These instructions may be transmitted in a near-real time/real-time configuration, in a just-in-time configuration, and/or provided ahead of time to allow for planned/scheduled routes. Further, in some embodiments, the command generation and scheduler module **316** coordinates the reservations and clearances to help a robot expeditiously navigate its way across a facility.

The command generation and scheduler module **316** may be configured to provide a command set comprising a single path, or one or more paths, and/or a number of operations to be performed at various locations. The command generation and scheduler module **316** provides these commands to the robot communications module **318** to be provided to the individual robots. In some embodiments, the command generation and scheduler module **316** pre-populates instructions for a particular robot – these instructions may then be provided to the robot through the

robot communications module **318** to be executed at a future time.

The robot communications module **318** may be configured to transmit information back and forth from the robots via the one or more base stations and the base station controller **206**. In some embodiments, the robot communications module **318** may communicate through the use of wireless signals. As indicated above, these instruction sets are not necessarily just-in-time, instruction sets may be sent for the coordination of future movements.

The robot communications module **318** may receive status reports from various robots. The robot communications module **318** may be implemented in various ways, such as utilizing synchronous, asynchronous, polling, push or pull methodologies. Further, various implementations may or may not include the use of communications "handshaking".

In some embodiments where there is no "handshaking", the communication systems may not guarantee message delivery, and lost packets may ensue. Potential advantages to such a system may include decreased bandwidth requirements and the submission of instructions on a best efforts basis. Various schemes may be implemented to minimize the impact of lost packets, such as timed rebroadcasting of instruction packets, sending overlapping instruction packets that contain parity information or other validation schemes, or other flow control and retransmission schemes.

In other embodiments of the invention, "handshaking" may be used to ensure that packets are received.

Commands to the robots may be issued ahead of the start time for an operation to be performed by a robot, and this time between the start time and the issuance of a command may be a configurable parameter.

In some embodiments, commands are repeated to the robots to ensure delivery, and the robots may provide acknowledgements that commands were received.

Where a message is not received before a scheduled start time, the robot may be configured to ignore the command and may return a status message indicating that the command was received too late. In this case, the robot control system **202** may be configured to cancel the existing reservations for the robot; and re-plan the tasking for the robot.

In some embodiments, the robot returns a regular status message that acknowledges the last command received (for example, by a command sequence number). The robot control system **202**, in some embodiments, may be configured such that no new command can be issued to a particular robot until the last command issued is acknowledged by the robot. If the command is not acknowledged by the robot after a particular period (e.g. a configurable a time-out period); the robot control system **202** may be configured to cancel the existing reservations for the robot. When (command) communication is re-established with the robot; the robot control system **202** re-plans the operation for the robot.

On receipt of each robot status message, the robot control system **202** may be configured to extend a robot's current movement clearance through the clearance module **312**.

The charge manager module **320** may be configured to develop a movement plan to recharge robots. The charge manager module **320** may be configured to estimate when robots will have a specified minimum charge, and ensure that all robots are able to charge at or before that point.

The alert/notification module **322** may be configured to provide an alert or notification to the control interface **302** when a potential issue has arisen, or based upon a pre-determined business rule, e.g. a predetermined number of clearances have been withheld due to conflicts.

The present system and method may be practiced in various embodiments. A suitably configured computer device, and associated communications networks, devices, software and firmware may provide a platform for enabling one or more embodiments as described above. By way of example, **FIG. 1** shows a generic computer device **100** that may include a central processing unit ("CPU") **102** connected to a storage unit **104** and to a random access memory **106**. The CPU **102** may process an operating system **101**, application program **103**, and data **123**. The operating system **101**, application program **103**, and data **123** may be stored in storage unit **104** and loaded into memory **106**, as may be required. The computer device **100** may further include a graphics processing unit (GPU) **122** which is operatively connected to the CPU **102** and to memory **106** to offload intensive image processing calculations from the CPU **102** and run these calculations in parallel with

the CPU **102**. An operator **107** may interact with the computer device **100** using a video display **108** connected by a video interface **105**, and various input/output devices such as a keyboard **115**, mouse **112**, and disk drive or solid state drive **114** connected by an I/O interface **109**. In a known manner, the mouse **112** may be configured to control movement of a cursor in the video display **108**, and to operate various graphical user interface (GUI) controls appearing in the video display **108** with a mouse button. The disk drive or solid state drive **114** may be configured to accept computer readable media **116**. The computer device **100** may form part of a network via a network interface **111**, allowing the computer device **100** to communicate with other suitably configured data processing systems (not shown). One or more different types of sensors **135** may be used to receive input from various sources.

The present system and method may be practiced on virtually any manner of computer device including a desktop computer, laptop computer, tablet computer or wireless handheld. The present system and method may also be implemented as a computer-readable/useable medium that includes computer program code to enable one or more computer devices to implement each of the various process steps in a method in accordance with the present invention. In cases of more than one computer device performing the entire operation, the computer devices are networked to distribute the various steps of the operation. It is understood that the terms computer-readable medium or computer useable medium comprises one or more of any type of physical embodiment of the program code. In particular, the computer-readable/useable medium can comprise program code embodied on one or more portable storage articles of manufacture (e.g. an optical disc, a magnetic disk, a tape, etc.), on one or more data storage partitions of a computing device, such as memory associated with a computer and/or a storage system.

The mobile application of the present invention may be implemented as a web service, where the mobile device includes a link for accessing the web service, rather than a native application.

The functionality described may be implemented to any mobile platform, including the iOS™ platform, ANDROID™, WINDOWS™ or BLACKBERRY™.

It will be appreciated by those skilled in the art that other variations of the

embodiments described herein may also be practiced without departing from the scope of the invention. Other modifications are therefore possible.

A number of different algorithms and techniques may be used in determining a preferential path for a robot to take, including, but not limited to: branch and bound algorithms, constraint programming, local search, heuristics, graph-traversal, dynamic path learning advisor techniques, pruning techniques and Bayesian graph searching techniques, Dijikstra's algorithm, Bellman-Ford algorithm, Floyd-Warshall algorithm, Johnson's algorithm, breadth-first recursive searches and depth-first recursive searches, weighted paths, A* search algorithm, variants on A* search algorithm (e.g. D*, Field D*, IDA*, Fringe, Fringe Saving A*, Generalized Adaptive A*, Lifelong Planning A*, Simplified Memory Bounded A*, Jump Point Search, Theta*).

In the sections below, sample search algorithms and heuristics are provided, according to some embodiments of the invention.

**Sample Search Algorithm**

In this section, a sample, simplified algorithm is provided, according to some embodiments of the invention.

For illustrative purposes, the algorithm is provided graphically as a workflow under **FIG. 4**, according to some embodiments of the invention. It is to be understood that this sample algorithm is a non-limiting example that is solely provided as illustration to the concepts as described above.

The algorithm may be an iterative search algorithm that may utilize a branch & bound search and may apply a "near-best-first" heuristic model including a 'heat map' for congestion avoidance. Branches may be selected using a weighted cost function, and the algorithm may be loosely coupled to grid/robot shape/size.

In some embodiments, branches may be held in a sorted collection.

According to an embodiment of the algorithm, the following recursive branch and bound function is applied:

On each iteration:

(a) Select lowest cost branch **b**,

(b) Branch **b** in all directions to create new branches **B**

(c) For each branch **b'** in **B**:

If **b'** has reached target:

return **b'**

Else:

Add **b'** to search

Cost comparator allows tracking of lowest cost branch for next iteration

In performing a path search, various heuristics may be applied to reduce the computations required, by, depending on the heuristic applied, removing entire branches or conducting a less computationally intensive analysis. Sample heuristic techniques are provided in a later section of this specification.

Where a new branch has a conflict with a path that another robot may be taking, or conflicts with an idle robot, the search algorithm may:

Alter the branch to outrun the conflicting reservation (if acceleration profile allows);

Alter the branch to contain a wait at any position along its path; or

Discard the branch as infeasible

In some embodiments, paths may be selected where robots wait at their starting points.

In doing so, the search space can never truly be exhausted (e.g. if there are currently no acceptable paths for a robot to take to its destination, a path may be selected where the robot waits until a non-conflicting path is available).

**Sample Heuristic Techniques**

The search algorithm may be configured to preferentially balance any number of goals such as taking the shortest possible time, tending to avoid congested areas etc.

The following provides a simple, non-limiting example of the application of the use of heuristics for illustrative purposes.

In an embodiment of the invention, each search may track the current lowest cost branch and weighted cost functions may be used to bias the selection ordering of the branches based upon various heuristics, which may include: **(a) a projected shortest path time** and **(b) a projected accumulated heat based on a heat-map**. Other heuristic techniques may be contemplated but for illustrative purposes, further detail will be provided for the two identified above.

**(a) Projected Shortest Path Time**

For any branch, the control system may determine the shortest possible path time from the current branch tip to the destination.

The projected time cost may then be determined as the total time for the branch so far (including, for example, any waiting that was required, etc.) added to the time for the shortest possible unconstrained path to the target.

**(b) Projected Minimum Heat (based on a heat map)**

In developing a heat-map, a 'heat' value may be assigned to each coordinate, approximating a model of congestion at points on the grid, or for particular areas of the grid.

**FIG. 5** provides a sample heat map, according to some embodiments of the invention.

In some embodiments, the 'heat' value may be determined using proximity to workstations, but in other embodiments of the invention, the 'heat' may be observed / learned / calculated / predicted using a variety of other techniques, some of which may be dynamic or iterative techniques.

Similar to the projected shortest path, an unconstrained path may be projected to the destination.

The projected minimum heat may then be determined. In some embodiments, the projected heat cost is the sum of the heat of all visited coordinates in the current branch added to the heat of the 'coldest' (the least hot) of the projected paths.

**Sample Weighted Cost Function**

In some embodiments, the algorithms used are based upon weighted cost functions. Such algorithms may be amenable to optimisation of the associated cost coefficients

by studying the results of large numbers of concurrent simulations in the cloud configured to use different coefficients, and/or applying various machine learning approaches and techniques, possibly using large sets of observed and/or simulated data.

In some embodiments, the search algorithm has two cost coefficients: (a) the projected shortest path time coefficient ($C_t$), and (b), the projected minimum heat coefficient ($C_h$).

In some embodiments, the search algorithm may include the following equation:

Branch cost = $C_t$*PSP + $C_h$*PMH

(Where PSP refers to **projected shortest path time** and PMH refers to **projected minimum heat)**

In some embodiments, the cost function may be utilized with configurable or machine-learning derived exponents that model complex relationships. A sample, simplified cost function, provided for illustrative purposes includes:

Branch cost = $C_t$*PSP$^x$ + $C_h$*PMH$^y$, where x & y may be further configurable and/or machine learned exponents.

**FIG. 6a** provides a table demonstrating how a search branch's projected cost could change as a result of using different coefficients in the path search algorithm. These coefficients could be derived using refinement by a machine learning technique according to some embodiments of the invention.

**FIG. 6b** illustrates how the use of different cost coefficients demonstrated in **FIG. 6a** changes which branch will be selected by the next iteration of the search algorithm.

In some embodiments, the control system **202** may be further configured to develop, adapt and apply a set of rules over time to refine a set of machine learned coefficients and/or exponents. The use of machine learned coefficients and/or exponents may potentially increase the effectiveness of the heuristic techniques over a duration of time.

**Sample Shunting Search**

The system may be configured to adjust robot paths to take into account the positions of idle robots. In some embodiments, there may be idle robots which may

be tracked independently of robots with tasks. These idle robots may not have planned paths and associated reservations and may need to be considered separately.

A separate "shunt search" may be performed when a task is being finalised. The "shunt search" may be comprised of finding paths to move robots which are idle now, or will go idle in the path of the robot which is being tasked (hereinafter referred to as the primary robot), to locations where they may continue being idle and not be in the way of the path of the primary robot.

This "shunt search", in some embodiments, comprises performing a search where, for each robot which is idle now, or will go idle in the path of a primary robot, a search is performed which may be deemed solved upon finding a location in which it can remain indefinitely.

The "shunt search" may use the same branch & bound search algorithm as the primary robot path search, but may have different cost coefficients and solution criteria. If a robot is unable to move out of the way in time, a wait may be added to the start of the primary robot's path and the primary robot's path may be recalculated.

**Sample Put-away Algorithm**

An algorithm may be used to determine a stack location for a container to be returned to. Containers may be returned for various reasons, and the location in which a container is returned to may be optimised for various advantages, such as improving the distribution of objects/containers in the hive.

Every stack location in the hive may be scored with a configurable, weighted cost function of:

Average distance (measured in robot operation time) from all workstations;

Distance (measured in robot operation time) from closest workstation; and

Approximate dig cost (if depth > 0)

The system may keep a "Hive Plan" of the current end state of the hive after all operations in the plan have been executed.

The "Hive Plan" may also track the "available surface" in which a robot can place a container. Each container has an index of its position in the totally ordered set of containers, as defined by the product placement optimisation module **306**.

Each stack location has an equivalent index in the totally ordered set of stack locations as defined by the weighted cost function.

These indices are remapped to the range 0-1 by dividing by the size of their respective sets, and the stack locations in the available surface are ranked by how closely their indices match that of the container's index.

The final selection is made via a weighted cost function of the difference between these indices and other factors such as how long the ideal path is from the source to the stack location and how long the stack is reserved for in the current plan.

Other business rules can be enforced at this stage, such as limiting the total weight of a stack; controlling the position of hazardous or special substances (e.g. aerosols and inflammable materials) etc.

**Sample Return Scenario**

The following provides a sample returns process that may be supported / controlled by the system. This process could be applied where an order is scratched, totes did not leave the hive, an order is returned by customer, or a delivery failed to a customer. Other situations may also be contemplated. A returned product (which may be in a container, other holding device such as a tote, etc.) can be processed at a workstation that provides reworking or rejection of the products.

The container/tote may be scanned so that the controller can position the storage bins it expects to need close to the workstation. Supply bins may be selected based on SKU and expiry date. Product items may be removed by the picker one-by-one and scanned. When the container arrives at the workstation, a picker (automated or manual) may be instructed / controlled to place the item into the container.

The picker may also be asked to confirm there are no further eaches of this SKU# left, before the container is released.

Products that are no longer suitable for return to stock may be picked into containers; which, at various times, such as when full, or at the end of the day, may be removed at the workstation and the contents sent to another area, such as the

staff shop or disposal as appropriate.

While the disclosure has been provided and illustrated in connection with specific, presently-preferred embodiments, many variations and modifications may be made without departing from the spirit and scope of the invention(s) disclosed herein. The disclosure and invention(s) are therefore not to be limited to the exact components or details of methodology or construction set forth above. Except to the extent necessary or inherent in the processes themselves, no particular order to steps or stages of methods or processes described in this disclosure, including the Figures, is intended or implied. In many cases the order of process steps may be varied without changing the purpose, effect, or import of the methods described. The scope of the invention is to be defined solely by the appended claims, giving due consideration to the doctrine of equivalents and related doctrines.

## CLAIMS

1. A system arranged to control movement of a plurality of transporting devices, the plurality of transporting devices arranged to transport containers, the containers being stored in a facility, the facility arranged to store the containers in a plurality of stacks, the facility comprising a plurality of pathways arranged in a grid-like structure above the stacks, the plurality of transporting devices arranged to operate on the grid-like structure, the system comprising:

a control means (202) arranged to determine at least one task to be performed by at least one of the plurality of transporting devices,

wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on at least one of: transporting device battery condition, transporting device damage, transporting device maintenance issues, and transporting device service issues.

2. The system according to Claim 1, wherein the at least one task comprises one of: moving from a first location on the grid-like structure to a second location on the grid-like structure, communicating with the control means (202), communicating with another transporting device, coordinating movement with another transporting device, transporting containers each of which may contain one of more items, lifting containers from the stacks, lowering containers into the stacks, delivering empty containers to inbound stations, placing containers loaded with goods around the facility, bringing containers to pick stations or other areas, and moving containers from one location in the facility to another.

3. The system according to Claim 1 or Claim 2, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device is malfunctioning.

4. The system according to any preceding claim, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device requires charging.

5. The system according to any preceding claim, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device can perform the task given a set amount of battery life.

6. The system according to any preceding claim, wherein the control means (202) is arranged to receive a status report from each of the plurality of transporting devices.

7. The system according to any preceding claim, the control means (202) comprising a movement control means (302) arranged to control movement of the plurality of transporting devices.

8. A storage system comprising:

a facility arranged to store containers in a plurality of stacks, the facility comprising a plurality of pathways arranged in a grid-like structure above the stacks;

one or more transporting devices arranged to operate on the grid-like structure and arranged to transport containers; and

a system according to one of the preceding claims.

9. A storage system according to Claim 8, wherein each transporting device has a footprint of a single grid position on the grid-like structure.

10. A method for controlling movement of a plurality of transporting devices, the plurality of transporting devices arranged to transport containers, the containers being stored in a facility, the facility arranged to store the containers in a plurality of stacks, the facility comprising a plurality of pathways arranged in a grid-like structure above the stacks, the plurality of transporting devices arranged to operate on the grid-like structure, the method comprising the step of:

determining at least one task to be performed by at least one of the plurality of transporting devices,

wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on at least one of: transporting

device battery condition, transporting device damage, transporting device maintenance issues, and transporting device service issues.

11. The method according to Claim 10, wherein the at least one task comprises one of: moving from a first location on the grid-like structure to a second location on the grid-like structure, communicating with a control means, communicating with another transporting device, coordinating movement with another transporting device, transporting containers each of which may contain one of more items, lifting containers from the stacks, lowering containers into the stacks, delivering empty containers to inbound stations, placing containers loaded with goods around the facility, bringing containers to pick stations or other areas, and moving containers from one location in the facility to another.

12. The method according to Claim 10 or Claim 11, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device is malfunctioning.

13. The method according to any of claims 10 to 12, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device requires charging.

14. The method according to any of claims 10 to 13, wherein the at least one task to be performed by the at least one of the plurality of transporting devices is determined based on whether the transporting device can perform the task given a set amount of battery life.

15. The method according to any of claims 10 to 14, the method further comprising the step of receiving a status report from each of the plurality of transporting devices.