

CLAIMS

1. A method for controlling vehicles to perform transport tasks comprising:
supplying information about the vehicles and information about the transport tasks to a graph neural network by associating each vehicle with a graph node of a vehicle graph and each transport task with a graph node of a transport task graph;
processing the vehicle graph and the transport task graph by the graph neural network, wherein the graph neural network is a neural network trained to determine a feature for each graph node of a graph it processes;
determining, for each pair of a graph node of the transport task graph and a graph node of the vehicle graph, a weight representing a similarity between the feature determined for the graph node of the transport task graph and the graph node of the vehicle graph;
selecting an assignment between graph nodes of the transport task graph and graph nodes of the vehicle graph from a set of possible assignments, wherein the selected assignment maximizes, among the possible assignments, a sum, over pairs of graph node of the transport task graph and graph node of the vehicle graph which are assigned to each other, of weights of the pairs; and
controlling each vehicle which is assigned to a transport task according to the selected assignment to perform the transport task,
wherein each feature is a vector of a predetermined dimension, and
the weight between the feature determined for the graph node of the transport task graph and the graph node of the vehicle graph is given by a vector inner product of the feature determined for the graph node of the transport task graph and the graph node of the vehicle graph.
2. The method of claim 1, wherein selecting the assignment comprises applying an assignment problem algorithm to a bipartite graph having the vehicle graph as first graph component, the transport task graph as second graph component,

and edges between the first graph component and the second graph component with the determined weights.

3. The method of claim 2, wherein the assignment problem algorithm is a min-sum or a max-sum algorithm.
4. The method of claim 2 or claim 3, wherein the assignment problem algorithm has a fixed number of iterations.
5. The method of any one of claims 2 to 4, wherein the assignment problem algorithm is differentiable.
6. The method of any one of claims 1 to 5, comprising supplying, for each vehicle, information about the vehicle as one or more input feature values for the graph node of the vehicle graph associated with the vehicle to the graph neural network, and, for each transport task, information about the transport task as one or more input feature values for the graph node of the transport task graph associated with the transport task to the graph neural network.
7. The method of claim 6, wherein the vehicle graph comprises edges between graph nodes depending on a similarity of the input feature values of the graph nodes and wherein the transport task graph comprises edges between graph nodes depending on the similarity of the input feature values of the graph nodes.
8. The method of any one of claims 1 to 7, wherein, for at least some of the vehicles, the information about the vehicles comprises location information of the vehicles.
9. The method of any one of claims 1 to 8, wherein each transport task comprises picking up an object of person to transport and, for at least some of the

transport tasks, the information about the transport tasks comprises location information about where the object or person needs to be picked up.

10. The method of any one of claims 1 to 9, comprising training the graph neural network using reinforcement learning.

11. A method for training a graph neural network comprising:
forming training data elements by, for each training element,
associating each vehicle of a training set of vehicles with a graph node of a vehicle graph for the training element, and each training transport task with a graph node of a transport task graph for the training element;
determining a label for the training element by determining a training assignment of the training set of vehicles to a training set of transport tasks;
and
training the graph neural network by
for each training data element

processing the vehicle graph and the transport task graph by the graph neural network;

determining, for each pair of a graph node of the transport task graph and a graph node of the vehicle graph, a weight representing a similarity between a feature determined for the graph node of the transport task graph and the graph node of the vehicle graph; and
selecting an assignment between graph nodes of the transport task graph and graph nodes of the vehicle graph from a set of possible assignments, wherein the selected assignment maximizes, among the possible assignments, a sum, over pairs of graph node of the transport task graph and graph node of the vehicle graph which are assigned to each other, of weights of the pairs,

wherein each feature is a vector of a predetermined dimension, and the weight between the feature determined for the graph node of the transport task graph and the graph node of the vehicle graph is given

by a vector inner product of the feature determined for the graph node of the transport task graph and the graph node of the vehicle graph; and adjusting the graph neural network to reduce a value of a loss function depending on a sum of differences between selected assignments and training assignments over the training data elements, wherein each difference is the difference between a selected assignment and a training assignment for a respective training data element.

12. A transport system controller configured to perform the method of any one of claims 1 to 11.
13. A computer program element comprising program instructions, which, when executed by one or more processors, cause the one or more processors to perform the method of any one of claims 1 to 11.
14. A computer-readable medium comprising program instructions, which, when executed by one or more processors, cause the one or more processors to perform the method of any one of claims 1 to 11.