Claims

1. A computer-implemented method for enabling parallel processing of determining collision-free paths for a vehicle to navigate from a start location to a target location, the method comprising:

receiving representations of the start location (SL), the target location (TL) and receiving one or more polygonal representations (CO1, CO2, CO3) of one or more potential collision objects;

initializing at least one representation of the vehicle, the at least one representation having the received start location (SL) as current location (CL), and further having a list of targets (LOT) comprising the received target location (TL) as most recent target location;

iteratively constructing a graph of collision-free path segments by determining whether a collision object is located on a direct connection between the current location and the most recent target location in the list of targets, and repeating for each vehicle representation:

if the direct connection is collision-free, saving the path segment as an edge of the graph, and selecting, for a next iteration, the most recent target location as new current location (CL) and removing the most recent target location from the list of targets;

else, identifying a plurality of reachable nodes wherein the plurality of reachable nodes are based on local extrema on one or more polygonal representations intersecting with the direct connection, where local extrema are determined relative to a deviation angle to the left or to the right from the direct connection between the current location and the most recent target location, and, for the next iteration, generating from the representation a further representation of the vehicle for each identified node, the further representation having the same current location as the vehicle representation and having a list of targets comprising all elements of the vehicle representation's list of targets as well as the identified node as new most recent target location;

until, the lists of targets associated with the existing vehicle representations are empty; and

constructing collision-free paths based on collision-free path segments; and

providing the collision-free paths to a path selector.

2. The method of claim 1, wherein identifying the plurality of reachable nodes further comprises:

determining a first group of polygon segments of a particular polygonal representation which comprises the polygonal segments with the deviation angle to the left for circumventing a corresponding collision object, and determining a second group of

polygon segments of the particular polygonal representation which comprises the polygonal segments with the deviation angle to the right for circumventing the corresponding collision object;

for each determined polygon segment of the first group, independently marking the node having the higher value in the deviation angle to the left, and for each determined polygon segment of the second group, independently marking the node having the higher value in the deviation angle to the right; and

identifying nodes with two marks as local extrema of the particular polygonal representation as potentially reachable nodes.

3. The method of claim 2, wherein identifying the plurality of reachable nodes further comprises:

identifying nodes with the highest values in deviation angles to the right and to the left as global extrema of the particular polygonal representation and filtering out local extrema which are not global extrema.

4. The method of claim 2, wherein identifying the plurality of reachable nodes further comprises:

filtering out local extrema of a particular polygonal representation which are hidden for the current location by one or more local extremum of a polygonal representation.

5. The method of claim 1, wherein identifying the plurality of reachable nodes further comprises:

identifying nodes with the highest values in deviation angles to the right and to the left as global extrema of the particular polygonal representation and filtering out local extrema which are not global extrema; and

filtering out local extrema of a particular polygonal representation which are hidden for the current location by one or more local extremum of a polygonal representation.

6. The method of claim 1, wherein identifying the plurality of reachable nodes based on local extrema is performed in parallel by a multi-processor device.

7. The method of claim 1, wherein constructing the graph further comprises one or more of the following performance based filtering steps:

filtering local extrema by checking the admissibility of intermediate target locations; and

deleting a particular vehicle representation if the particular vehicle representation can be judged to perform worse than a competing vehicle representation with regards to a respective goal function.

8. The method of claim 1, wherein the path selector applies one or more of the following post-processing activities to the constructed collision-free paths:

smoothening of a path by using splines;

further adding one or more waypoints along the path, wherein said way points represent entry or exit points to one or more areas of interest; and

removing one or more waypoints from the path if a vehicle can move directly from a preceding to a succeeding waypoint.

9. The method of claim 1, further comprising:

receiving, from the path selector, a selected collision-free path; and

updating a control system of the vehicle with the selected collision-free path to adjust the moving direction of the vehicle according to the selected path.

10. The method of claim 1, wherein a particular polygonal representation of a particular potential collision object is associated with sensor data indicating the current position of the particular potential collision object.

11. The method of claim 1, wherein, when constructing the graph and the direct connection is collision-free, deleting the representation of the respective vehicle if the list of targets is empty.

12. A computer program product enabled for parallel processing to determine collision-free paths for a vehicle to navigate from a start location to a target location, the computer program product, when loaded into a memory of a computer system and executed by at least one processor of the computer system, performs the steps of:

receiving representations of the start location (SL), the target location (TL) and receiving one or more polygonal representations (COL CO2, CO3) of one or more potential collision objects;

initializing at least one representation of the vehicle, the at least one representation having the received start location (SL) as current location (CL), and further having a list of targets (LOT) comprising the received target location (TL) as most recent target location;

iteratively constructing a graph of collision-free path segments by determining whether a collision object is located on a direct connection between the current location and

the most recent target location in the list of targets, and repeating for each vehicle representation:

if the direct connection is collision-free, saving the path segment as an edge of the graph, and selecting, for a next iteration, the most recent target location as new current location (CL) and removing the most recent target location from the list of targets;

else, identifying a plurality of reachable nodes wherein the plurality of reachable nodes are based on local extrema on one or more polygonal representations intersecting with the direct connection, where local extrema are determined relative to a deviation angle to the left or to the right from the direct connection between the current location and the most recent target location, and, for the next iteration, generating from the representation a further representation of the vehicle for each identified node, the further representation having the same current location as the vehicle representation and having a list of targets comprising all elements of the vehicle representation's list of targets as well as the identified node as new most recent target location;

until, the lists of targets associated with the existing vehicle representations are empty; and

constructing collision-free paths based on collision-free path segments; and

providing the collision-free paths to a path selector.

13. The computer program product of claim 12, wherein the at least one processor of the computer system is further operable to identify a plurality of reachable nodes by performing the further steps of:

determining a first group of polygon segments of a particular polygonal representation which comprises the polygonal segments with the deviation angle to the left for circumventing a corresponding collision object, and determining a second group of polygon segments of the particular polygonal representation which comprises the polygonal segments with the deviation angle to the right for circumventing the corresponding collision object;

for each determined polygon segment of the first group, independently marking the node having the higher value in the deviation angle to the left, and for each determined polygon segment of the second group, independently marking the node having the higher value in the deviation angle to the right; and

identifying nodes with two marks as local extrema of the particular polygonal representation as potentially reachable nodes.

14. The computer program product of claim 13, wherein identifying the plurality of reachable nodes further comprises:

identifying nodes with the highest values in deviation angles to the right and to the left as global extrema of the particular polygonal representation and filtering out local extrema which are not global extrema.

15. A computer system for parallelized determination of collision-free paths for a vehicle to navigate from a start location to a target location, comprising:

an interface component configured to receive start and target location data and one or more polygonal representations of one or more potential collision objects;

an initialization component configured to initialize at least one representation of the vehicle, the at least one representation having the received start location as current location, and further having a list of targets comprising the received target location;

a graph generator configured to iteratively construct a graph of collision-free path segments by iteratively calling an extrema identifier module and a replicator module for each vehicle representation as long as the received target location is not reached, to determine whether a collision object is located on a direct connection between the current location and the most recent target location in the list of targets, wherein:

the graph generator is configured to save a corresponding path segment as an edge of the graph if the direct connection is collision-free, and is further configured to select, for the next iteration, the most recent target location as new current location and to remove the most recent target location from the list of targets;

the graph generator being further configured, if the direct connection is not collision-free,

to call the extrema identifier for identifying a plurality of reachable nodes wherein the nodes are based on local extrema on one or more polygonal representations intersecting with the direct connection, where local extrema are determined relative to the deviation angle to the left or to the right from the direct connection between the current location and the most recent target location, and

to call the replicator to generate, for the next iteration, from the representation a further representation of the vehicle for each identified node, the further representation having the same current location as the vehicle representation and having a list of targets comprising all elements of the vehicle representation's list of targets as well as the identified node as new most recent target location;

the graph generator further configured to construct collision-free paths based on collision-free path segments; and to provide the collision-free paths to a path selector.

16. The computer system of claim 15, wherein the extrema identifier is further configured to identify the plurality of reachable nodes by:

determining a first group of polygon segments of a particular polygonal representation which comprises the polygonal segments with the deviation angle to the left for circumventing the corresponding collision object, and determining a second group of polygon segments of the particular polygonal representation which comprises the polygonal segments with the deviation angle to the right for circumventing the corresponding collision object;

for each determined polygon segment of the first group, independently marking the node having the higher value in the deviation angle to the left, and for each determined polygon segment of the second group, independently marking the node having the higher value in the deviation angle to the right; and

identifying nodes with two marks as local extrema of the particular polygonal representation as potentially reachable nodes.

17. The computer system of claim 15, wherein identifying the plurality of reachable nodes further comprises:

identifying nodes with the highest values in deviation angles to the right and to the left as global extrema of the particular polygonal representation and filtering out local extrema which are not global extrema.

18. The computer system of claim 15, wherein the extrema identifier is further configured to identify the plurality of reachable nodes by:

identifying the reachable nodes with the highest values in deviation angles to the right and to the left as global extrema of the particular polygonal representation; and

obtaining the reachable nodes by filtering out the local extrema which are hidden for the current location by the particular polygonal representation.

19. The computer system of claim 15, wherein the computer system is a mobile device.

20. The computer system of claim 15, wherein, when constructing the graph and the direct connection is collision-free, deleting the representation of the respective vehicle if the list of targets is empty.