

June 2016. Messe Leipzig, Available online at www.sciencedirect.com

IFAC-PapersOnLine 49-15 (2016) 212–217

3D Navigation Mesh Generation for Path Planning in Uneven Terrain Planning in Uneven Terrain $\frac{30}{20}$ New Approximative $\frac{30}{20}$ Path $\frac{30}{20}$ 3D Navigation Mesh Generation for Path 3D Navigation Mesh Generation for Path Planning in Uneven Terrain

Sebastian Putz ∗ Thomas wiemann ∿ochen Sprickerhof
T_{ess}ibine Hutzber#*** Joachim Hertzberg ∗∗∗∗ Sebastian Pütz ∗ Thomas Wiemann ∗∗ Jochen Sprickerhof ∗∗∗ Sebastian Pütz ∗ Thomas Wiemann ∗∗ Jochen Sprickerhof ∗∗∗ Joachim Hertzberg ∗∗∗∗ Joachim Hertzberg ∗∗∗∗ Sebastian Pütz ∗ Thomas Wiemann ∗∗Jochen Sprickerhof ∗∗∗ Joachim Hertzberg ∗∗∗∗

∗ *Osnabrück University, Germany (spuetz@uni-osnabrueck.de)* ∗∗ *Osnabrück University, Germany (twiemann@uni-osnabrueck.de)* ∗ *Osnabrück University, Germany (spuetz@uni-osnabrueck.de)* ∗∗ *Osnabrück University, Germany (twiemann@uni-osnabrueck.de)* ∗∗∗ *Osnabrück University, Germany (jspricke@uni-osnabrueck.de)* ∗∗ *Osnabrück University, Germany (twiemann@uni-osnabrueck.de)* ∗∗∗ *Osnabrück University, Germany (jspricke@uni-osnabrueck.de)* ∗∗∗∗ *Osnabrück University and DFKI Robotics Innovation Center,* ∗∗∗ *Osnabrück University, Germany (jspricke@uni-osnabrueck.de)* ∗∗∗∗ *Osnabrück University and DFKI Robotics Innovation Center, Osnabrück Branch, Germany (joachim.hertzberg@uni-osnabrueck.de)* ∗∗∗∗ *Osnabrück University and DFKI Robotics Innovation Center,* ∗ *Osnabrück University, Germany (spuetz@uni-osnabrueck.de) Osnabrück Branch, Germany (joachim.hertzberg@uni-osnabrueck.de)* ∗∗∗∗ *Osnabrück University and DFKI Robotics Innovation Center,* ∗ *Osnabrück University, Germany (spuetz@uni-osnabrueck.de) Osnabruck University, Germany (jspricke@uni-osnabrueck.de) • Osnabruck University and Dr N1 Robotics Innovation Center*

Osnabrück Branch, Germany (joachim.hertzberg@uni-osnabrueck.de)

a 3D point cloud to reconstruct a triangle mesh of the environment in real time that is enriched with a graph structure to represent local connectivity. This *Navigation Mesh* is then analyzed for roughness and trafficability and used for online path planning. The presented approach is evaluated with a VolksBot XT platform in a real life outdoor environment. Abstract: We present a 3D mesh surface navigation system for mobile robots. This system uses evaluated with a volksbot Λ i platform in a real life outdoor environment. a 3D point cloud to reconstruct a triangle mesh of the environment in real time that is enriched with a graph structure to represent local connectivity. This *Navigation mesh* is then analyzed for roughness and traincaphity and used for online path planning. The presented approach is

© 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved. *Keywords: Robot Navigation, Path Planning, Surface Reconstruction, Mesh Generation, M* © 2016 IFAC (International Federation of Automatic Control) Hosting by Elsevier Li

Keywords: Robot Navigation, Path Planning, Surface Reconstruction, Mesh Generation *Keywords:* Robot Navigation, Path Planning, Surface Reconstruction, Mesh Generation *Keywords:* Robot Navigation, Path Planning, Surface Reconstruction, Mesh Generation

1. INTRODUCTION 1. INTRODUCTION 1. INTRODUCTION 1. INTRODUCTION

In flat terrain most obstacles can be safely perceived by In flat terrain most obstacles can be safely perceived by using horizontal 2D laser scanners. Compared to that, $\frac{1}{2D}$ laser scanners. Compared to that, it is much more difficult to operate in rough terrain and to check the trafficability of uneven ground. Robots of unknown environments of unknown environments or exploration operating outdoors like in rescue operations or exploration operating outdoors like in rescue operations or exploration
of unknown environments need to cope with such uneven or unknown environments need to cope with such uneven
terrain and must still be able to find a path to their goal. terrain and must sum be able to find a path to their goal.
Therefore, they have to classify if something is a potential Therefore, they have to classify if something is a potential
obstacle or if they can pass it, e.g., by traversing over $\frac{1}{2}$ it. To reach the target, the trafficability of the ground between the current position and the goal pose is essential. Rough terrain involves certain risks, such as holes in the Rough terrain involves certain risks, such as holes in the
ground, bold peaks or sharp objects. There might be heavy ground, bold peaks of sharp objects. There inight be heavy
obstacles that pose danger to the robot and that must be obstactes that pose danger to the robot and that must be
avoided in any case. But there might also be obstacles avoided in any case. But there imgin also be obstacles the robot can deal with at higher costs (energy, execution time) in order to reach the target. time) in order to reach the target. and to check the trafficability of uneven ground. Robots obstacle or if they can pass it, e.g., by traversing over it. To reach the target, the trafficability of the ground between the current position and the goal pose is essential. the robot can deal with at higher costs (energy, execution time) in order to reach the target.

We have developed a general mesh planner extending the we have developed a general mesh planner extending the
ideas of Breitenmoser and Siegwart (2012) and Gingras et al. (2010). This paper presents our approach for local planning and robot control in the current field of view and planning and robot control in the current held of view and
demonstrates the usability on a mobile robot operating in a real world outdoor environment. Unlike 2D grid map a rear world of 2010 of 2D grid map approaches, this 3D planner works on triangle meshes approaches, this 3D planner works on triangle meshes
instead of 2D grid map planes. Working with such 3D instead of 2D grid map planes. Working with such 3D
surfaces in navigation has the advantage of conformity of surfaces in havigation has the advantage of comornity of
the navigation costs based on distances, height differences, meshes represent all and space of the space of the space directly and space directly roughness, forbidden areas and other criteria. Additionally, notations, increased and can enter the relationship, mesnes represent 3D space unectly and can easily be an-
notated with semantic labels as described in Deeken et al. (2014) . Compared to 2.5D solutions, full 3D mesh surface (2011). Compared to 2012 botations, ran of most surface navigation algorithms enable havigating enough a matter ever environment, such as multistory buildings of tunner
systems. In particular mesh navigation is appropriate for climbing robots or other robots with the ability to access walls or ceilings. walls or ceilings. ideas of Breitenmoser and Siegwart (2012) and Gingras et al. (2010). This paper presents our approach for local demonstrates the usability on a mobile robot operating in a real world outdoor environment. Unlike 2D grid map the navigation costs based on distances, height differences, roughness, forbidden areas and other criteria. Additionally, notated with semantic labels as described in Deeken et al.
(9914) $C = 1 + 95D - 1 + 65D - 1 + 6D - 1 + 6D$ (2014). Compared to 2.5D solutions, full 3D mesh surface navigation algorithms enable navigating through a multi systems. In particular mesh navigation is appropriate for climbing robots or other robots with the ability to access walls or ceilings.

In this paper we present an open source, robot-independent m the paper we present an open solute, reset imagenation.
3D mesh navigation solution for robot navigation with high extensibility. high extensibility. In this paper we present an open source, robot-independent In this paper we present an open source, robot-independent 3D mesh navigation solution for robot navigation with high extensibility.

2. RELATED WORK 2. RELATED WORK 2. RELATED WORK

2.1 Octree Mapping and Digital Elevation Maps 2.1 Octree Mapping and Digital Elevation Maps 2.1 Octree Mapping and Digital Elevation Maps 2.1 Octree Mapping and Digital Elevation Maps

Many navigation and path planning approaches use Ocwhany navigation and path planning approaches use Ocpoint cloud data. A popular implementation for it is the point cross state. The popular imprementation for the end
OctoMap package in ROS (Hornung et al., 2013). A planoccorring passage in 1000 (riormang octan, 2019). It plant regions that are located that are locally and the used of all (2009) . It decomposes the 3D Octree map into several (2009). It decomposes the 3D Octree map mto several
regions that are locally 2D. These regions can than be used regions that are locally 2D. These regions can than be used
like a ordinary grid map for path planning. Also, semantic the determining given map for pair prairing. These, commuted classification of the represented surfaces can be integrated to speed up the computation. Fig. 1b shows an example of the Octree representation of a natural scene. to speed up the computation. Fig. 1b shows an example of classification of the represented surfaces can be integrated the Octree representation of a natural scene. the Octree representation of a natural scene. Many navigation and path planning approaches use Oc-Many navigation and path planning approaches use Ocone octree representation or a natural section point cloud data. A popular implementation for it is the OctoMap package in ROS (Hornung et al., 2013). A planning system for Octrees is described in Morisset et al. like a ordinary grid map for path planning. Also, semantic to speed up the computation. Fig. 1b shows an example of the Octree representation of a natural scene.

Occupancy maps, like Octrees, represent obstacles in a Occupancy maps, intercretes, represent obstactes in a
binary way in form of a grid map. These maps only define obstacles, but not the trafficability of obstacles in a specific obstactes, but not the traincability of obstactes in a specific
context. For example, an obstacle can provide drivable context. For example, an obstacle can provide dirvable
ground, if the robot is on top of it, but also an inaccessible ground, it are research on top or to, such and the observed ledge, if the robot is located at a lower level. Generally, distance to the obstacle as a following constant, as evaluated in Schwarz and the obstacle detection depends on the obstacle's size and the obstacle detection depends on the obstacle s size and the distance to the obstacle, as evaluated in Schwarz and distance to the obstacle, as evaluated in Schwarz and
Behnke (2014). To deal with these shortcomings, elevation behinke (2014) . To deal with these shortcomings, elevation maps build on top of simple occupancy maps. Starting maps sum on top or simple socapancy maps. Starting
with height maps, where a height z is stored in the cells with height maps, where a height z is stored in the central of the grid map with $z = f(x, y)$. These solutions are of the grid map with $z = f(x, y)$. These solutions are usually called 2.5D maps (cf. Fig. 1c). A special type usually called 2.5D maps (cl. Fig. Ic). A special type
of 2.5D maps are *Digital Elevation Maps* (DEM). These maps are used to compute local trafficability maps, e.g., maps are used to compute focal traincability maps, e.g.,
by calculating the navigation costs based on the height by calculating the havigation costs based on the height differences on multiple scales (Schwarz and Behnke, 2014). differences on multiple scales (Schwarz and Bellike, 2014).
However, these techniques are not able to handle multiple overlapping levels, such as tunnel systems or multistory overlapping levels, such as tunnel systems or multistory However, these techniques are not able to handle multiple Occupancy maps, like Octrees, represent obstacles in a binary way in form of a grid map. These maps only define ground, if the robot is on top of it, but also an inaccessible
ledge if the relation legated at a lemma level. Commetter ledge, if the robot is located at a lower level. Generally, maps build on top of simple occupancy maps. Starting of 2.5D maps are *Digital Elevation Maps* (DEM). These overlapping levels, such as tunnel systems or multistory

2405-8963 © 2016, IFAC (International Federation of Automatic Control) Hosting by Elsevier Ltd. All rights reserved. **Peer review under responsibility of International Federation of Automatic Control.** 10.1016/j.ifacol.2016.07.734 **Co Copyright Copyright Copyright Copyright Contract Copyright 10.1016 /; if each 2016 07.724**

(a) Lab example scene with a step field.

(b) Octree representation of the scene.

(c) 2.5D Map (DEM) representation of the scene.

(d) Mesh representation of the scene.

Fig. 1. State of the art map representations to estimate trafficability in 3D in uneven terrain.

buildings and are not practicable for climbing robots. For evaluation we implemented the DEM trafficability estimation schemes presented in Schwarz and Behnke (2014) in a *costmap_2d* layer plugin for *move base* in *ROS*. An example DEM costmap is shown in Fig. 1d together with the reconstructed mesh for comparison.

2.2 Terrain Reconstruction

An alternative approach is described in Gingras et al. (2010) and Breitenmoser and Siegwart (2012). This method is able to handle multiple levels by reconstructing the rough terrain into a triangle mesh. Such triangle mesh representations are much more flexible and can be used for multiple purposes like environment representations in robotic simulators (Wiemann et al., 2013), visualization for human robot interaction and ground classification or path planning (Wiemann et al., 2010).

In such a representation, the alignment of surface parts or change between adjacent triangles can be evaluated for drivability estimation. This allows to determine navigation costs based on distances, height difference, roughness, forbidden areas and semantic information.

3. SYSTEM ARCHITECTURE

The components of our approach are shown in Fig. 2. The main input is a triangle mesh constructed with some kind of surface reconstruction procedure. In principle, arbitrary meshes from different sources like LVR (Wiemann et al., 2012) or PCL (Rusu and Cousins, 2011) could also be used. In this paper, we use the *Organized Fast Mesh* method to generate a triangle mesh surface representation from 3D laser scanner data at the current position of the robot in real time (Holz and Behnke, 2012). In this paper we do not present an approach for global mapping. However, our approach also works for meshes representing larger environment sections, coequal with some kind of a global map.

First, a so called *Navigation Mesh* is generated and analyzed to estimate distances, height differences, and roughness. If specific safety thresholds are violated, these areas will be marked as lethal obstacles, since the robot must avoid them. Areas around such lethal obstacles are inflated (marked as dangerous) and graph optimization algorithms are applied, e.g., *Graph Edge Region Growing* to implicitly smooth the discretized mesh path. Now it is possible to apply shortest path algorithms on the resulting optimized *Navigation Mesh* to compute paths taking the evaluated

Fig. 2. Structure of our terrain classification approach. Triangle mesh representations are converted into a graph that is used for planning using navigation costs estimated for the represented terrain.

criteria into account. Besides global navigation, the *Navigation Mesh* can be used for a local control of the robot. Details on the single computation steps are presented in the following sections.

4. SURFACE RECONSTRUCTION

4.1 Organized Point Clouds

The organized fast meshes we use in this paper rely on organized point clouds. The data flow to generate them is displayed in Fig. 3. After removing outliers from a single 2D laser scan and applying a robot self filter, we assemble the single 2D laser scans to one organized point cloud. In such a representation, neighboring points are stored in a matrix structure resulting in a depth image. To generate this matrix, we exploit the fact that we know the polar angles ϕ and θ for every single point. Our rotation unit delivers for every single point the ϕ value. The θ value can be reconstructed from the order of points in each 2D scan taking the point count and the resolution into account. In summary: The Cartesian coordinates (x, y, z) of a scan point corresponding to the spherical coordinates ϕ and θ neighboring scan line distance values are stored in a matrix structure. This structural knowledge about the data allows us to connect neighboring points in a single 360 degree scan to construct the mesh. Unsensed or removed points are marked with *NaN* to maintain the matrix structure. Another advantage of the organized structure is that it enables a fast and robust computation of point normals by using integral image techniques such as *Average 3D Gradient*, *Average Depth Change* or *Covariance Matrix* as described in Holzer et al. (2012).

Download English Version:

<https://daneshyari.com/en/article/708732>

Download Persian Version:

<https://daneshyari.com/article/708732>

[Daneshyari.com](https://daneshyari.com)