RAVON — The Robust Autonomous Vehicle for Off-road Navigation

C. Armbrust, T. Braun, T. Föhst, M. Proetzsch, A. Renner, B.H. Schäfer, and K. Berns*

Abstract

With the aim of developing a vehicle that can fully autonomously operate in highly vegetated terrain, the University of Kaiserslautern’s Robotics Research Lab is conducting research in the field of off-road robotics. Their platform RAVON is a 4WD vehicle equipped with a large number of different sensor systems, which are used as basis for the hazard detection.

RAVON’s navigation system is three-layered, consisting of a deliberative navigator on top of a behaviour-based pilot. The navigator’s job is to create and update topological maps of the robot’s environment, in which paths are calculated. The pilot tries to move the robot along such paths while keeping it away from obstacles. Collision avoidance is realised by a large number of behaviours that access the sensor data in a unified and straightforward way. An intermediate layer shall mediate between navigator and pilot and solve problems they cannot address properly.

The purpose of this paper is to provide information about the platform RAVON and to present the concepts underlying the work in detail.

1 Introduction

In April 2003, the Robotics Research Lab of the University of Kaiserslautern was founded by Prof. Dr. Karsten Berns. Five years later, 17 research associates belong to the lab. The focuses of their work include autonomous off-road robotics and behaviour-based approaches.

Collision-free mobile robot navigation belongs to the most difficult tasks in the development of robotic systems. This especially applies to navigation in highly unstructured, harsh, and dangerous environments (see section 1.1 for examples). Such environments make great demands on a vehicle’s hardware as well as its software. Its kinematics must allow for conducting sophisticated manoeuvres on different types of terrain, its sensor systems must detect a large variety of hazards, and its navigation system must be able to deal with complex, possibly dangerous situations.

With the use of robot’s for risky applications in mind and with the desire to solve fundamental scientific problems in the field of off-road robot navigation, the Robotics Research Lab has started the development of an entirely autonomous vehicle that is able to operate safely in rough and

*The authors are with the Robotics Research Lab at the Department of Computer Sciences, University of Kaiserslautern, P.O. Box 3049, 67653 Kaiserslautern, Germany (e-mail: {armbrust, braun, foehst, proetzsch, renner, b_schaef, berns}@cs.uni-kl.de).
highly vegetated terrain. Their experimental platform is RAVON, the Robust Autonomous Vehicle for Off-road Navigation (see figure 1).

A sophisticated hazard detection based on different sensor systems, a reliable obstacle avoidance, and a flexible and versatile high-level navigation in complex environments have been identified as key features of their research platform.

![Figure 1: The autonomous off-road robot RAVON.](image)

1.1 Motivation

During the recent years, the demand for robotic vehicles has strongly increased in various domains, including unmanned space travel [Sherwood 01; Tunstel 01; Schenker 03], automatic gathering of measurements [Ray 05], archaeological exploration [Gantenbrink 99], agricultural automation [Thuilot 01; Lenain 03; Wellington 04], and evolution of military devices [Hong 02; Zhang 01; Debenest 03].

Unmanned vehicles could patrol borders, guard industrial estates, fulfil reconnaissance tasks in hostile environments, or assist in clearance duties in cases of severe accidents or natural disasters. In all of these scenarios, the risk for people’s lives could be reduced by providing them with information about a previously unclear situation, or even by keeping them completely out of potentially dangerous situations. But despite the strong need for autonomous vehicles, most of the implementations still appear more like enhanced remote-controlled cars [Kuni 01] than autonomously acting and decision-taking, thus “intelligent”, robots.

While the task of going from one place to another is usually simple for a human, it can be far from trivial for an autonomous robot. Even in highly structured environments (cp. section 2.1), there are still many problems to be solved before robotic vehicles can operate fully autonomously without posing a threat to their environment. Therefore, it is generally regarded as safer to let humans take most of the decisions necessary for a robot to accomplish its mission.

One crucial drawback of this approach (which is referred to as “semi-autonomous”) is the fact that radio communication often suffers from limited bandwidth and connection interruptions due to obstacles between robot and operator. During the ELROB 2008, many of the teams relying on a permanent radio connection between their vehicle and their base station had to abort when the connection got lost.

1ELROB: European Land-Robot Trial ([http://elrob.org/](http://elrob.org/))
Apart from this problem, high bandwidth communication with the control headquarters can consume a considerable amount of the energy the robot carries around. Depending on the application, recharging might be time-consuming, complicated, or even impossible. A Mars rover, for example, cannot simply drive to the next power outlet to recharge its accumulators. Robotic vehicles are often equipped with electric actuators, which also need energy. Although in most cases combustion engines would solve the energy problem satisfyingly [Fukushima 01], there are many reasons for nevertheless banking on battery-powered systems. Augmented noise, weight, and pollution shall suffice to be mentioned here. So as energy is usually scarce, it should certainly not be wasted for keeping up a radio link to an operator.

Furthermore, remotely deciding what actions are safe is not always easy from the sensory data provided by a mobile robot. Most of the time this will be video images, which are normally 2D, limiting depth perception. In general, a lot of training will be necessary in order to master vehicle control in difficult situations. Introducing more sophisticated perceptional systems and preprocessing units will probably render remote-controlling even more complex.

Recapitulatory, current remote control mechanisms result in long operation times and high costs for qualified personal. Both issues shall be addressed with self-dependent navigation systems which enable mobile robots to find their way to a given target area fully autonomously.

### 1.2 Structure of this Work

The following section will give an overview of the state of the art in the wide field of autonomous outdoor navigation, with the focus lying on the work dealing with the problems of off-road navigation. Sections 3 and 4 will deal with the autonomous off-road robot RAVON. The different hardware components will be presented and the main concepts underlying RAVON’s control system will be explained. The purpose of section 5 is to discuss the current state of the project on the basis of expressive experiments. It will lead over to concluding remarks and an outlook on future work, which are presented in section 6. Finally, section 7 contains acknowledgements to Team RAVON’s sponsors.

### 2 State of the Art

Worldwide, various research projects deal with autonomous navigation of a robot in outdoor terrain. However, the target environments of these approaches differ significantly in terms of structuredness and types of obstacles. The projects can be separated into those targeting on urban environments and those dealing with non-urban terrain. The latter can be divided further into projects aiming at on-road and off-road environments, respectively.

This section will give an overview of the state of the art in the field of outdoor robot navigation by presenting relevant projects.

#### 2.1 Navigation in Urban Terrain

Many of the projects in the field of outdoor robotics focus on urban environments. Though fully autonomous cars are far from being available, a number of systems known as advanced driver assistance systems have been developed and integrated into standard cars during the last years. Such systems include adaptive cruise control and automatic parking. They increase safety or comfort by providing partial autonomy that can help the driver in certain situations.
The success of these systems and the attention drawn to the field by the DARPA Urban Challenge\(^2\) motivate researchers to put effort into the work on robots that autonomously navigate in urban environments. Research topics include road sign detection\(^3\), road following\(^4\), and pedestrian detection (see \cite{Zhao00} and figure \ref{fig:Navlab} for work of the Navlab group and \cite{Navarro-Serment08} for work that is actually not focused on urban environments).

However, the differences between urban and rural or even off-road terrain are large and so the results of this research area can only be partly applied to the field of off-road robotics.

**Figure 2:** Boss, the winning vehicle of the DARPA Urban Challenge.

**Figure 3:** Navlab 11.

### 2.2 Off-road Navigation in Terrain with Little to No Vegetation

The projects which belong to this category often target on autonomous navigation in desert-like areas (see \cite{Urmson04}). This has been fostered by the DARPA Grand Challenges held in 2004 and 2005, in which vehicles had to follow autonomously a route defined by GPS points (see, e.g., \cite{Thrun06} and figure \ref{fig:UrbChal}). The focus of these challenges lay on high-speed (over 50km/h) navigation in a terrain with minor jaggedness and only sparse vegetation. Furthermore, the route followed a path that could be recognised using vision systems in order to assist the navigation.

Another huge field of application for autonomous navigation is the space exploration sector. Needless to say, vegetation is no problem in this application. However, the terrain is often dominated by jagged rocky formations and there are also hazards like holes in the ground and abysms. As a consequence, the approaches published by NASA\(^5\)/JPL\(^6\) aim at this type of terrain. The approach for visual terrain mapping described in \cite{Olson07} is even more specialised: It does not only use surface images from a rover, but also images taken by a spacecraft as it approaches the planetary surface, and images from spacecrafts orbiting a planet.

A robot system which features elaborate facilities for local obstacle detection, localisation, and navigation is introduced in \cite{Lacroix02}. Yet the computational effort only allows for very low velocities (about 10cm/s).

Due to the high signal propagation delay between a control station on the earth and a spacecraft, telecommanding is difficult and solely relying on it increases the risk of accidents that can damage or even destroy the vehicle. Some approaches try to overcome these problems by providing a rover with a set of waypoints which it shall follow autonomously between communication cycles (see \cite{Ravine07}). However, the current state of the art in this field is the use of little autonomy.

\(^2\)\url{http://www.darpa.mil/grandchallenge/}
\(^3\)NASA: National Aeronautics and Space Administration (\url{http://www.nasa.gov/})
\(^4\)JPL: Jet Propulsion Laboratory (\url{http://www.jpl.nasa.gov/})
2.3 Off-road Navigation in Highly Vegetated Terrain

Only few research projects deal with the problem of autonomous robot navigation on highly vegetated terrain. The different types of hazards in such environments put high demands not only on a robot’s navigation strategies, but also on the algorithms used for sensor processing.

Of all the projects in this category, a large number have a military background. Among them is the DEMO III project, whose goal was to advance and demonstrate the technology which is required to develop future unmanned ground combat vehicles that can operate autonomously in cross-country vegetated terrain (see [Shoemaker 98] and figure 6). In the context of this project, many approaches for obstacle detection and terrain classification have been developed and investigated. Several contributions about such approaches have been made by researchers of the JPL (see [Bellutta 00], [Talukder 02], [Manduchi 05]). Other work dealt with architectural concepts of the control system (see [Albus 02]). Discriminatory for the DEMO III project (and other projects with military background) is the large variety of expensive technology like LADAR, high-resolution stereo systems and powerful computational units as well as the semi-automated system architecture. The system does not operate fully autonomously, but is supported by a human operator in exploring, navigating, as well as tackling difficult situations.

More recent work about obstacle detection in highly vegetated terrain includes processing point clouds generated from LADAR data (see [Vandapel 04]), building up a probabilistic terrain model in order to improve ground estimates and obstacle detection (see [Wellington 06]), and the use of red and near-infrared reflectance of obstacles to discriminate between vegetation and other obstacles (see [Bradley 07]). But still, the problem of hazard detection in off-road terrain is far from being solved.
3 Hardware

This section shall present the hardware of the autonomous mobile off-road robot RAVON. The first part will provide information about the basic platform, while the second part will present the robot’s various sensor systems.

3.1 Basic Platform

The basis of RAVON is a robot platform manufactured by Robosoft, called robuCAR TT (see figure 7). As the original platform did not meet the high demands on an off-road platform’s stability and endurance, many parts have been reinforced or replaced by the members of the Robotics Research Lab, among them the mountings for the shock absorbers and the transverse links. Several sensor systems, computers, and other components have been added, resulting in the current version of the robot RAVON, which is depicted in figure 8.

Including all components, RAVON measures 2.4 m by 1.4 m by 1.8 m (length × width × height) and weighs 750 kg, which makes it comparable to a small car in terms of size and weight (see figure 9).

Figure 7: robuCAR TT, the platform manufactured by Robosoft.

Figure 8: RAVON during the ELROB 2008.

Figure 9: A technical drawing of RAVON (top view).
RAVON’s two axes can be steered independently, which supports the robot’s agility and allows for conducting advanced manoeuvres like double Ackermann steering. Thus RAVON can make sharp turns and increase the distance to lateral obstacles without changing the robot’s orientation. Many tasks like clearance duties after a severe accident or a natural disaster require such a high agility as there are typically many obstacles around which the vehicle has to manoeuvre.

RAVON is powered by four independent electric motors that have been kindly sponsored by Johannes Hübner Giessen[^5]. Each of them has a power of 1.9kW and is controlled by a high performance motor controller provided by Unitek Industrie Elektronik[^6]. These controllers are connected to the main PC via CAN-bus. With this equipment, RAVON can reach a maximum velocity of 3 m/s, and in combination with its Hankook off-road tires it can climb slopes of up to 100%.

RAVON is equipped with four industrial PCs assembled and sponsored by DSM Computer[^7]. They are responsible for running the high-level control systems for local navigation (Intel Core2Duo T7400), global navigation (Intel Core2Duo E4300), as well as low-res image processing and data collecting (Intel Pentium M 780 and Intel Core2Duo E6700). Five DSP boards developed at the Robotics Research Lab are attached to the computers using two CAN buses. They are running low-level programmes that realise e.g. the motor control.

The power for all of RAVON’s systems is delivered by 8 OPTIMA YT S 4.2 accumulators sponsored by OPTIMA Batteries[^8]. Each of them has a capacity of 55Ah. This suffices for an operation time of 3 to 4 hours, depending on factors like driving speed and steepness of the terrain.

A custom casing has been built up of MiniTec[^9] profile elements to protect the electronic components from dirt and humidity. The upper part can be opened to the back so that RAVON’s interior can be accessed easily.

Although RAVON is designed to operate fully autonomously, for testing purposes a wireless connection is needed in order to be able to send commands to the robot and receive telemetry data. Therefore a standard WLAN connection is available. In order to establish connections over longer distances, RAVON is equipped with a universal data transceiver produced by IK Elektronik[^10]. In combination with a 500mW amplifier distances of up to 10km can be bridged at a data rate of 115kbit/s. An appropriate omnidirectional HF aerial has been mounted onto a gimbal to ensure vertical alignment of the antenna at severe slopes (see figure[^10]).

### 3.2 Sensor Systems

In order to be well prepared for the large variety of situations RAVON can get into, it is equipped with a number of different sensor systems (see figure[^9]). They can be separated into two groups: sensor systems that estimate the robot’s pose (i.e. position and orientation) and sensor systems designed for gathering information about the robot’s environment.

An encoder is attached to each of RAVON’s four motors. They are used as basis for a pose calculation using wheel odometry. As the robot often operates on slippery or uneven terrain, this only yields a rough estimate of the true pose, so other components are integrated into the estimation

[^5]: http://www.huebner-giessen.com/
[^6]: http://www.unitek-online.de/
[^7]: http://www.dsm-computer.de/
[^8]: http://www.optimabatteries.com/
[^9]: http://www.minitec.de/
[^10]: http://www.ik-elektronik.com/
process. A custom inertial measurement unit (IMU) that measures movements along and rotations around all three axes (see [Koch 05]) and a magnetic field sensor are mounted at the upper rear part of the casing (see figure 11). Furthermore, a John Deere[11] StarFire iTC (see figure 13) and a u-blox[12] AEK-4H (see figure 14) are employed as absolute position sensors. The estimates delivered by these sensors are combined using a Kalman filter (see section 4.3).

![Figure 10: The HF aerial.](image1)

![Figure 11: Cube of inertial measurement unit and magnetic field sensor.](image2)

![Figure 12: One of RAVON’s bumpers.](image3)

![Figure 13: John Deere StarFire iTC.](image4)

![Figure 14: u-blox AEK-4H.](image5)

RAVON is equipped with three SICK[13] laser range finders. Two LMS 291 are attached to the lower front and rear of the robot, respectively (see figure 15). An LMS 291 has a field of vision of 180°, an angular resolution of 0.5°, and a distance resolution of 0.5 cm. The disadvantage of a fixedly mounted laser scanner is that it can only detect obstacles which intersect its scan plane. Relying solely on such sensors may be sufficient in very simple, structured environments. However, the highly unstructured environments in which RAVON shall operate demand for more sophisticated obstacle detection capabilities. Therefore, a third SICK laser scanner (an S300) is mounted in a custom mounting bracket that can pan the scanner to the left and to the right. As can be seen in figure 15 this scanner is attached to a sensor tower built up of MiniTec profile elements. By mounting the scanner with its scan plane upright and panning it continuously, 3D information about the robot’s environment can be gathered. Using this data, RAVON’s sensor processing system is able to detect hazards like water, overhanging obstacles and holes in the ground. How this is done is explained in section 4.2. In the current state of the system, the panning laser range finder is the main sensor system delivering information about the terrain on which the robot moves.

Two stereo vision systems are also mounted on the sensor tower. The lower of the two consists of two Point Grey Research Dragonfly2. It can be used for obstacle detection or visual odometry. The cameras are mounted on a pan/tilt unit so that they can monitor a large area in front of the robot. The higher of the two stereo vision systems consists of two high resolution cameras (Point Grey Research SCOR-20SOC-CS CCD). It is used by the global navigation system to gather detailed terrain information at certain navigation points. These two cameras are also mounted on as pan/tilt unit, which is placed on the highest position of the sensor tower. Hence the cameras have a good view on a large part of the area around the robot.

![Figure 15: RAVON’s sensor tower.](http://www.mayser.de/)

Two spring-mounted safety bumpers manufactured by Mayser are attached to the front and rear, respectively (see figure[12]). In a special operation mode called “tactile creep”, the spring system is used to separate soft obstacles from rigid ones in situations where the geometric obstacle detection alone cannot be used. If one of the bumpers collides with an obstacle that cannot be pushed away easily, the robot’s safety chain is opened, which results in an immediate stop of the vehicle.

The sensor systems described here allow for hazard detection in a variety of environments. Depending on the application scenario, other sensors have to be added. For example, special sensors for detecting chemical substances would be needed if the robot shall be used after an accident in a chemical plant. For patrolling or guarding tasks, additional sensors for detecting humans may be needed.

4 Software

This section’s purpose is to provide information about the general ideas and concepts behind RAVON’s control software. A high-level overview of its components is depicted in figure[16]

---

[14]: http://www.mayser.de/
4.1 Framework

All four of RAVON’s industrial PCs run Gentoo Linux. The robot control software has been implemented using the Kaiserslautern branch of the C++ robot control framework MCA\textsuperscript{15}. This framework was originally developed at the FZI\textsuperscript{16} in Karlsruhe. It is now being developed under the name MCA-KL at the University of Kaiserslautern’s Robotics Research Lab. The KL branch is a strongly extended version with a large number of additional features and user libraries. It is used on all of the robots developed at the Robotics Research Lab. There are also MCA components that do not run on a PC, but on the DSPs used on custom-design DSP boards (e.g. one for RAVON’s IMU), which allows for preprocessing sensor data before it is sent to a PC. MCA-KL, further libraries, and tools are published under the GPL\textsuperscript{17}.

The behaviour-based architecture upon which parts of RAVON’s control system are based (see section 4.4) is called iB2C\textsuperscript{18}. It is a modified derivative of the architecture introduced in \cite{Luksch 02} and \cite{Albiez 03}. The modifications made are partly exemplified in \cite{Proetzsch 05}. Performance and flexibility of the approach have been demonstrated in numerous case studies, one of which is presented in \cite{Schäfer 05a}.

The current version of the iB2C incorporates several concepts which facilitate the structuring of large systems and the reuse of components. Among these concepts are special behavioural groups that combine other behaviour-based components and generic behaviour templates that can be used in the control systems of different robots by adding robot-specific elements. The iB2C architecture has been implemented using C++ and has been integrated seamlessly into MCA-KL.

The fundamental unit of the proposed control architecture is the behaviour module (see figure \ref{fig:ib2c}). Each atomic behaviour is wrapped into such a module which computes the meta output signals \textit{activity} and \textit{target rating}. The impact of behaviours on the overall control of the robot can be influenced using their meta inputs \textit{stimulation} and \textit{inhibition}. The meta signals allow the arrangement of behaviours in a comprehensive hierarchical fashion, which supports the extension

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{control_system_diagram.png}
\caption{A high-level overview of the control system’s components.}
\end{figure}

\textsuperscript{15}MCA: Modular Controller Architecture
\textsuperscript{16}FZI: Research Center for Information Technology \url{http://www.fzi.de/}
\textsuperscript{17}\url{http://rrlib.cs.uni-kl.de/}
\textsuperscript{18}iB2C: integrated Behaviour-Based Control
of the control system without touching existing behaviours and interconnections. As already al-
luded above, related behaviours can be wrapped into behavioural groups (see figure 18), which 
comply to the same interface as behaviour modules. That way another hierarchical level can be in-
troduced to manage complex behaviour networks. More information about the iB2C can be found 
in [Proetzsch 07].

Figure 17: The interface of a be-
haviour module, includ-
ing activity $a$, target rat-
ing $r$, stimulation $s$, and 
inhibition $i$.

Figure 18: An example of a behavioural group.

4.2 Sensor Processing

RAVON features various sensor systems that provide different views on its environment, in which 
different properties can be detected depending on the specific sensors’ capabilities. The bumper 
system, for example, yields information about traversable ground or solid obstacles by sweeping 
through the environment along with the robot’s movements. This manifests in the property of be-
ing known as traversable space for every position that was covered during the operation time of 
the robot. Another example is the panning laser scanner, which gathers three-dimensional infor-
mation about the robot’s frontal environment. It can distinguish between objects of different sizes 
and altitudes and thus discriminate between rough ground and severe obstacles like big stones or 
overhanging obstacles, e.g. rigid branches (see figure 19 and [Schäfer 08a]). Based on this sen-
sor it is also possible to guess if detected obstacles are rigid or soft vegetation by sampling the 
recorded data into a discrete grid and performing statistical analyses along with a voxel penetra-
tion technique (ray-tracing). This allows for proceeding at low speed and pushing through high 
gras, etc.

Figure 19: Different types of obstacles that can be detected and classified using the panning laser 
range finder: a) positive obstacle, b) positive step, c) positive slope, d) overhanging 
obstacle, e) negative obstacle, f) negative step, g) negative slope
To decouple the development of the sensor systems from problems on higher levels of the control system, the capabilities of the sensor and the direct processing of their data, i.e. filtering or feature extraction, define the set of “properties” that can be assigned to the observed environment by this particular sensor or its processing system. These properties are used to fill a local short-term memory implemented in form of a grid map with a specific resolution and range for each sensor (figure 20 and figure 22 a)). The following systems along with their grid maps have been implemented in the current version of the system:

- The bumper memory that stores whether a part of the environment is traversable by marking the driven path in its short-term memory.
- The 2D proximity detection which memorises whether solid objects were detected within the ranges of the lower 2D laser range finders.
- The 3D obstacle detection which memorises the different properties that were extracted from the raw recordings of the panning laser range finder.
- The filtered 3D obstacle detection which memorises the different properties that were extracted from the preprocessed recordings of the panning laser range finder that were cleansed of readings which probably belong to vegetation.
- The water detection which uses a statistical analysis of void readings from the panning laser range finder to mark places in its short-term memory that are probably covered by or consisting of water.
- The stereo camera based obstacle detection that was implemented on RAVON and described in [Schäfer 05b]. It is not in use at the moment due to the research on laser-based 3D obstacle detection.

As this representation is strictly predetermined by the development of the sensor systems, an abstract view for further processing is needed. Thus, a generic data structure that facilitates accessing the data of various sensors by providing a uniform interface called sector map [Armbrust 07] is used.
The underlying grid maps provide fast access to areas defined by simple geometric objects like lines or convex polygons. Using these, it is easy to define areas and relevant properties within a grid map of a specific sensor. Area and property definitions are used to generate sector maps that divide the regions they cover into several sectors. The sector maps then store for each sector whether it contains one of the given properties and, if yes, the distance from the sector’s origin to the most relevant representative for the current configuration (see figures 21 and 22 b)).

The mapping from the sensors’ grid maps and properties to the abstract view in the sector maps is formulated in terms of aspects and can be used to break down the information needed for different control aspects into a set of relevant properties of appropriate sensors. Using this virtual sensor layer, an action/perception-based design of RAVON’s control software is realised [Schäfer 08b].

Figure 22: An example configuration of RAVON’s virtual sensor layer: a) shows the short-term memory of the 3D obstacle detection and b) shows the resulting sector maps under the control aspect of avoiding to steer into the walls to the left and right.
4.3 Localisation

For many applications of robots, a good pose estimation is necessary. In indoor environments, SLAM approaches are often used (see, for example, [Eliazar 05]). However, indoor environments are typically highly structured, which reduces the noise in the data delivered by distance sensors. Furthermore, indoor robots usually do not need a three dimensional pose—their position on the ground and their orientation along the vertical axis is sufficient. On off-road terrain or in areas where a severe accident occurred, by contrast, a full three dimensional pose is often needed. Especially the roll and pitch angles are important in order to detect critical situations. Although there are approaches dealing with the application of SLAM in outdoor environments (see [Montemerlo 03]), this is difficult in extremely noisy environments or on terrain with few features like grassland and clearings.

Therefore, a fault-tolerant system architecture has been chosen for RAVON’s control system which can cope with dynamic changes in sensor availability and fulfils the requirements for a robust and adaptable solution (see [Schmitz 06a, Schmitz 06b]). The system is based on a linear Kalman filter and a flexible model of the system and sensors. It integrates the data provided by the vehicle’s odometry, the IMU including the magnetic field sensor, and one of the GPS devices. A switching between the two devices is realised to account for the different characteristics of the devices (high robustness and imprecise position estimation vs. lower robustness, but much more precise position estimation). The visual odometry is not integrated in the current version of the localisation system and is left to future work.

4.4 Local Navigation

The local navigation component of RAVON (the pilot) deals with assuring the basic safety requirements in respect to collisions. It directly utilises the virtual sensor layer to benefit from the uniform and abstract representation which allows for behaviour module reuse as well as straightforward integration of new sensors or sensor data processing algorithms.

![Figure 23: Some of the virtual sensors that are used by RAVON’s anti-collision system.](image)

Hence, the following characteristics can be differentiated:

- Arbitrary sensor coverage can be realised by extracting the corresponding region from the local grid map, e.g. front, rear, and side regions, with obstacles at different height levels.
• Several sensor data processing algorithms can be provided: raw laser scanner data evaluation, voxel penetration methods (for vegetation discrimination), water detection, etc.

• Different properties can be represented: rigid, soft, overhanging obstacles, holes (i.e. negative obstacles), water.

In order to meet the basic safety requirements, the first level (in a bottom-up design process) consists of behaviours that can trigger an emergency stop based on bumper events, overhanging or negative obstacles, as well as rigid obstacles in sensor height. As these obstacle types might harm the robot hardware, all drive commands issuing from higher layers are blocked.

The next layer consists of reactive behaviours for slowing down, rotating away from obstacles, and sideward motion for obstacle avoidance. As RAVON traverses scenarios with different characteristics, three drive modes (“control-aspects”) are defined. Each of them is represented by a behavioural group (see section 4.1). These groups are instances of the same class, provided with sensor information suitable for the respective drive mode:

• Fast driving requires long-range obstacle detection and the evaluation of all detected obstacles.

• Moderate drive through vegetation is provided with data from the voxel penetration evaluation for vegetation density determination. The resolution of the sensor systems only allows for the detection of close-range areas (up to 3m). Therefore, the maximum velocity is limited accordingly.

• Dense vegetation requires additional sensors as optical sensing cannot discriminate between solid and flexible objects. Therefore, the tactile creep mode issues a very slow drive command and supervises the deflection of the bumper system. In the case that rigid obstacles are hidden in the vegetation, the vehicle stops and backs off again. Furthermore, overhanging obstacles that might damage the other sensor systems result in the same reaction.

The three behavioural groups are stimulated by corresponding drive mode behaviours which implement the state switching using the concept of iB2C inhibitions. The safety behaviours gradually overwrite higher level commands by inhibition and by providing corrective motion both for velocity and steering commands.

The safety behaviour network is provided with drive commands originating from operator input behaviours, point access behaviours, as well as behaviours for preferring open space, driving along structures, etc.

The behaviour-based approach has several advantages that make it interesting for robots that shall operate in the scenarios described in section 1. The reactive aspect facilitates fast reactions on changing sensor data, while the modular structure makes a behaviour-based anti-collision system like the one implemented on RAVON tolerant to failures of a single sensor system. The behaviours operating on data of the failing sensor stop working, but the others proceed with their normal operation.

4.5 Global Navigation

The global navigation layer of RAVON (the navigator) is responsible for the generation of navigation decisions that have a spatial extent significantly larger than the immediate sensor horizon of the
robot. Thus, it concentrates on robot navigation tasks starting above a range of about 10 metres, going up to tasks which typically span several kilometres. The navigator deliberately abstracts from local aspects of the environment or the robot’s exact trajectory. These issues are to be handled by the local piloting subsystem that maintains a metrically more accurate, but spatially limited world representation.

### 4.5.1 Cost-Conscious Topological Path Planning

The navigator relies on a primarily topological map to store information about the connectivity of navigation-relevant places. In order to add the ability to perform cost-efficient path planning and map extension on the topological level, the initially unannotated topological map has been extended with several new aspects. These additions include a multi-dimensional cost measure for topological edges which records the ‘risk’, ‘effort’ and ‘familiarity’ cost aspects of each topological edge. In short, the risk cost factor quantifies the amount of evasive actions required by the pilot in order to traverse the edge. Likewise, the effort factor records the energy needed for edge traversal. Finally, the familiarity value is a virtual cost which quantifies the amount of cost knowledge already accumulated for each edge. Its purpose is to allow explicit influence on the robot’s explorative behaviour, e.g. its desire to traverse either along well known or new paths.

In order to determine cost values which are consistent with the real terrain properties without the need to analyse the terrain extensively and construct a highly detailed world model, a technique to learn such consistent cost values from scratch based on feedback from the robot’s pilot during operation has been developed [Braun 08a]. Figure 24 exemplarily shows two traces of the pilot’s spatially accumulated driving and obstacle avoidance behaviours (shown as green and red dots), which are observed by the navigator and ultimately integrated to form risk and effort costs.

Based on these estimates, a method to use the gained multi-dimensional cost information for path planning with user-selectable priorities has been proposed.

Figure 25 shows the result of an experiment which has been conducted to evaluate the effect of cost learning on path planning in a large real world scenario. The figure shows a panoramic image of the test area, overlaid with the topological map that was provided to the robot prior to cost learning. The testing ground covers approximately 100 by 100 metres and exhibits a maximum height difference of about 7 metres. The steepest part of the testing grounds (slope more than 20°) is located around node 3, while the most problematic obstacle configurations are below the bridge around nodes 1 and 2.

To obtain the path with shortest metrical distance, the path planner was commanded to plan a path from node 13 to node 7 using the unannotated, initial map. The result is marked in the figure with thick, white arrows. Then the robot was issued several dozens of random edge traversal commands in order to build up cost estimates. After cost learning, the path planner was requested to generate the connection between nodes 13 and 7 that minimises either the risk or effort cost sum. The resulting paths are indicated in the picture with dashed green (minimal risk) and dashed yellow (minimal effort) edges. Both paths differ substantially from the purely distance-based path. The yellow path saves energy by exploiting the steepest slopes around node 3 and using a relatively direct connection. This comes at the price of traversing difficult terrain around node 1. In contrast to this, the green path contains a lot of lengthy detours in order to avoid this area and the vicinity of the hedge (11–13). Both paths are intuitively plausible in the context set by the cost measure.

This experiment and further validation in simulation proves the claim that the layered navigation design and the proposed extensions of the topological map allow the navigator to build a minimal
Figure 24: Cost learning based on pilot behaviour traces.

<table>
<thead>
<tr>
<th>Cost Estimates (μ, σ)</th>
<th>Cost Estimates (μ, σ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Risk</td>
<td>(4.9, 3.8)</td>
</tr>
<tr>
<td>Effort</td>
<td>(669.0, 372.6)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Main Risk Contributors</th>
<th></th>
<th>Main Risk Contributors</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Avoid Left</td>
<td>54 %</td>
<td>Avoid Right</td>
<td>44 %</td>
</tr>
<tr>
<td>Avoid Right</td>
<td>24 %</td>
<td>Evasion</td>
<td>31 %</td>
</tr>
<tr>
<td>Evasion</td>
<td>22 %</td>
<td></td>
<td>20 %</td>
</tr>
</tbody>
</table>

Figure 25: Fish-eye view of testing area and overlaid topological map.
world model which is compact enough to be easily scalable up to large environments. By refining the initially coarse cost estimates continuously based on self-observation of the pilot, a more and more consistent measure of all cost-relevant aspects of the complex rugged off-road environment is obtained. With this approach, the additional knowledge invested into the global navigation layer can be kept at a minimum, and a high degree of robustness against sensor noise or inaccurate self-localisation can be achieved.

4.5.2 Topological Cost Prediction

To allow the prediction of traversability costs for topological edges which have not been traversed yet, a set of methods to extrapolate edge costs from existing cost information has been developed. Three of these methods reuse information stored in the cost annotations of the topological map edges, but incorporate data on different levels of locality. The coarsest prediction technique relies on a global cost model constructed from all available cost annotations using an outlier-robust linear regression of risk and effort cost factors. This model correlates the estimated travel length of the hypothetical edge with its probable cost by extrapolating global, overall terrain cost characteristics. A local cost model is built by the second prediction method, which restricts the spatial extent of edges eligible for cost transfer. This allows for better modelling local fluctuations of terrain properties, improving risk and especially effort predictions. The third approach predicts costs based solely on an edge’s direct inverse twin, which is the spatially closest source of information available.

In order to account for terrain properties which are not captured by the extrapolation of topological map information, a fourth cost prediction algorithm was proposed which uses ‘local traversability maps’ attached to the topological nodes as information source. These metrical maps store cost modifiers for possible exploration directions in the vicinity of topological nodes in a compact form (fig. 26).

![Figure 26: A local traversability map.](image)

The map is composed of radially arranged seclets. The cost modifier of each seclet is visualised using colours: green indicates low costs, yellow medium and red high cost modifier scores.

These cost modifiers are the key to predict the traversal costs of edges that lead into up to now unknown terrain. By combining the learned costs in the topological edges and the modifiers stored in the local metrical map, the accuracy of cost prediction can be greatly improved. This
is especially relevant for the risk cost factor, which depends on the amount of obstacle evasions during edge traversal.

Three strategies to fill local traversability maps have been proposed. First, the local obstacle memory of the pilot is examined and traversable free space and untraversable obstacles are extracted from it. Second and third, two sophisticated image analysis techniques have been proposed to fill additional traversability maps \cite{Braun08b} \cite{Braun08c} \cite{Zolynski08}. They use a long-range stereo camera system to estimate terrain traversability either based on surface shape (see figure 27a) or visual appearance (see figure 27b). A temporary terrain model is constructed and immediately abstracted into light-weight cost modifiers which are stored in a local traversability map. This ensures the maintenance of a minimal world model, leading to a scalable navigation approach which can be used in very large environments.

![Shape-based Analysis](image1)

(a) Shape-based Analysis

![Appearance-based Analysis](image2)

(b) Appearance-based Analysis

Figure 27: Image analysis methods to fill local traversability maps.

Green areas indicate analysis results that are translated into low cost seclets, red areas mark high cost seclets.

Experimental validation of the cost prediction algorithms revealed that increasingly accurate extrapolation techniques can be selected as the available cost information accumulates. The overall performance of the prediction strategy was verified and each method was quantitatively analysed based on an extensive large-scale simulation test.

Based on the developed cost extrapolation methods, an exploration strategy was proposed which generates new connection hypotheses from two sets of possibilities. The first set of extension hypotheses adds direct connections between a reachable node and the previously unreachable goal node. The second extension strategy inserts additional detour nodes, which are placed according to the sector sizes of the local traversability maps. This allows for optimally exploiting the traversability information contained inside. After evaluation of all valid hypotheses, only the candidate with the lowest predicted costs is actually incorporated into the map. This keeps the map as small as possible, in accordance to the formulated objective to retain a compact and scalable world model.

### 4.6 Navigation at an Intermediate Layer

While the interaction of the navigator and the pilot described above works well in simple environments, it can easily fail in more complex situations like the one depicted by figure 28. The reason
for this is that the collision avoidance works locally and misses the “big picture” of the current situation. For example, it cannot keep the navigator from drawing the robot away from the path and into the small opening to the right of the path. Such openings are so wide that the collision avoidance does not get active until the robot has driven into them. However, they could be easily recognised as indentations when using a scope that is larger than the one of the collision avoidance, but has a finer granularity than the world model used by the navigator.

![Diagram of RAVON facing a complex situation](image)

**Figure 28:** RAVON faces a complex situation while driving towards its target.

A robot can easily get into even more complex situations if it shall be used for clearance duties in case of a severe accident. In such environments, debris may block the robot’s path and leave only little space for moving.

In order to deal with situations that require analysing the big picture and improving the interaction between navigator and pilot, an intermediate navigation layer shall be realised between the two. A first part of this layer has already been developed: a component that searches for so-called passages, which are defined here as paths leading through obstacles. The sources for the detection process are virtual sensors represented by sector maps (cp. section 4.2) as they provide an easy and uniform access on the sensor data. A special type of virtual sensors is used to gather information about a passage’s length and orientation. The pose of such a sensor is not fixed in terms of the robot control system, but attached to some point of interest. As this resembles placing a sensor somewhere in the environment, this special type of virtual sensor is referred to as **virtual sensor probe**. Based on the collected data, passages are evaluated with respect to their value for the robot’s navigation. If a passage has been classified as relevant for the navigation, the commands of the navigator are overwritten by a special component that guides the robot through the passage. In the example, this makes the robot pass the indentation and drive straight ahead. Of course, the problem caused by the dead-end is not solved by this yet.

In the context of future work, further components shall be added to the intermediate navigation layer.

5 Experiments

Numerous experiments have been conducted with the robot RAVON in different environments in order to test the system and prove the effectiveness of the concepts presented here. Two test runs shall be presented in the following.
Figure 29: Using a passage to support navigation, RAVON keeps away from the indentation.

Figure 30: The views of a real sensor and a virtual sensor probe.

5.1 Experiment 1: Trial for Testing the Collision Avoidance

The main purpose of this test run was to validate the correct operation of the components dealing with hazard detection and collision avoidance. It was conducted in the Palatinate Forest around Kaiserslautern. RAVON was given a target in terms of a GPS coordinate that was located at approximately 1km air-line distance from the starting point. Intermediate waypoints were neither provided nor generated by the high-level navigation system (see section 4.5). Thus the robot was continuously drawn in the direction of the target, while the low-level behaviour-based navigation system (see section 4.4) avoided collisions based on the obstacle data provided by the sensor processing system (see section 4.2). The result was an autonomous movement towards the target.

Figure 31: The pose trace of experiment 1. The two checkpoints are marked with numbers.

Figure 32: The contents of the local obstacle memory and images taken at the two checkpoints.

Figure 31 shows a part of the complete test run. The blue line indicates the path of the robot estimated by the localisation system (see section 4.3). The test run was conducted with an older version of the hazard detection and avoidance system, which classified obstacles as critical (marked with red symbols) or non-critical (marked with green symbols) depending on the threat they pose to the robot. Note that RAVON’s control system does not generate a global metric map of its
environment. Therefore, the pose trace has been created offline on the basis of sensor and pose data recorded during the test run.

The situation at two checkpoints (marked with numbers in figure 31) shall be presented in detail. Figure 32 depicts the contents of the local obstacle memory at these checkpoints. The rays going to obstacles symbolise the contents of some relevant sector maps used by the anti-collision system. Objects displayed with green symbols are potentially negotiable, i.e. the robot may be able to pass over them. Obstacles marked with red symbols, by contrast, are critical, i.e. a collision has to be avoided by any means. A grey ray, finally, represents a sector without an obstacle and indicates the last detected ground point.

5.2 Experiment 2: Trial for Testing the Water Detection

If autonomous robots shall be used after a flood disaster, a reliable water detection is indispensable. In order to test RAVON’s water detection system, several test runs were performed. A significant one shall be presented here.

As the laser range finder is continuously moving, several pan movements are necessary to discover the whole water hazard. The process of water detection was observed in a map, while RAVON was driven towards different puddles. During the run, waters of many different sizes and depths were tested. Furthermore, the influence of the cleanness of the water was analysed.

![Figure 33: Some puddles from RAVON’s point of view and the corresponding map.](image)

The realized water detection was able to detect almost every water hazard on the path of the robot. Only shallow clear puddles smaller than 30cm in diameter remained undiscovered. In contrast to this, slightly muddy puddles were detected without any difficulty. The higher degree of diffusion caused by the small particles swimming in the water explains the observed behavior. This leads to the conclusion that in case of unclear water the depth of the hazard has no influence to the detection algorithm.

As depicted in figure 33, the sensor system was able to recognise water hazards in distances up to 9m. This detection range allows for a sufficiently foresighted control of the vehicle concerning obstacle avoidance.

5.3 Experiment 3: Trial for Testing the Passage Detection

In this experiment, the operation of the component which uses passages to support navigation (see section 4.6) was tested. It has also been conducted in the Palatinate Forest. Again, RAVON was given a target several hundred metres away from the starting point and had to get there fully autonomously. Figure 34 shows a short pose trace of the run. Obstacles are marked in a similar
Passages that have been detected are symbolised by three circles connected by a line. The colour is determined by the estimated quality of a passage (blue: best; green: suitable to support the navigation). The lines starting at the middle of the three circles provide information about a passage’s orientation. The red line starting at the robot indicates the direction in which the robot was drawn by the high-level navigation, i.e. the direction to the target.

At checkpoint 0 (see figure 35), the direction of the path leading through obstacles deviated much from the direction to the target. The draw of the high-level navigation would have led the robot directly into the underwood to its right. But as two suitable passages were detected, the robot did not follow this draw, but stayed on the path in order to pass through the passages.

A similar situation is depicted in figure 36 (checkpoint 1). Again, following the drag of the high-level navigation would have resulted in the robot driving into the underwood. As a result, the collision avoidance behaviours would have stopped the robot if it had gotten too close to the obstacles to its left. Then backing off would have been necessary. But by detecting and using passages, these time-consuming manoeuvres could be avoided.
6 Conclusion and Future Work

This paper gave a high-level overview of the autonomous off-road robot RAVON. The pronounced goal of the RAVON project—to develop a vehicle which is capable of driving fully autonomously through highly vegetated, difficult terrain—poses a number of requirements on the hardware as well as the software.

Due to the large variety of hazards in the application environments, different sensor systems had to be installed on RAVON, which have different characteristics and monitor different areas around the robot. A local obstacle memory is employed to keep track of obstacles when they leave the areas monitored by the sensors.

For the local navigation, a behaviour-based approach has been chosen as it allows for fast reactions to changing sensor data and can be easily modified or enhanced. Its main task is to realise collision avoidance and point access manoeuvres, while the global navigation provides a coarse driving direction and keeps track of the overall progress in reaching the target. Together, the two layers build a powerful navigation system.

In the current state, RAVON is able to navigate autonomously in a large variety of environments. Numerous experiments have been conducted on grassland and in the forest to prove the effectiveness of the proposed concepts. These experiments confirm that RAVON could be used as platform that can carry various devices needed in the scenarios named in section 1.

However, tests have also shown that the control system cannot cope with complex situations and certain types of hazards. Therefore, the sensors and the sensor processing system will have to be enhanced in the future. For example, the perception abilities of RAVON should be extended by additional sensors or processing algorithms, to detect muddy terrain or debris. For the use in risky environments, special sensors will have to be added in order to detect hazards like toxic chemical substances. Improvements in the virtual sensor layer will provide abstract sensor fusion on the higher level of sector maps. In contrast to merging raw sensor data, merging sector maps does not yield problems in terms of different sampling rates, resolution and semantic context. Furthermore, fusion in the abstract layer allows for more complex action design.

Additional research will go into large scale terrain classification to improve the navigator’s abilities to plan in the “big picture”. Points of interest are object modelling/fitting for place-recognition and extracting hints from the large scale view of the surrounding terrain for path-planning.

Furthermore, the intermediate navigation layer has to be extended. The detection and use of passages to improve navigation are a first step, but there are lots of complex situations for which concepts have to be developed on how to deal with them. These situations include RAVON reaching a dead-end on its path, which requires turning around and finding another way. All this is necessary for robustly navigating in harsh terrain and in environments with many hazards that are a potential threat to a robot for risky interventions.

7 Acknowledgements

Team RAVON thanks the following companies for their technical and financial support: IK elektronik, Mayser, Hankook, MiniTec, SICK, DSM Computer, Hübner Giessen, John Deere, Optima, ITT Cannon, MOBOTIX, and Unitek Industrie Elektronik.
References


