Optimal operation of sensing in multi-agent applications

Reza Ghabcheloo[a], Kalevi Huhtala[a], Mika Hyvönen[a], Heimo Ihalainen[b], Juha Koljonen[b], Antti Kolu[a], Mikko Lauri[b], Joonas Melin[b], Robert Piche[b], Jukka-Pekka Raunio[b], Aino Ropponen[b], Risto Ritala[b,]

[a]Department of Intelligent Hydraulics and Automation; [b]Department of Automation Science and Engineering; Tampere University of Technology, P.O. Box 692, FIN-33101 Tampere, Finland

KEY WORDS: robotics, sensing, machine cooperation, planning, communication

ABSTRACT

This paper presents the elements of sensing optimization. We concentrate on robotics applications and present the hardware, sensing capabilities, communication, state estimation and planning optimization methods. Three research environments are presented. Utilizing a micro aerial vehicle (MAV) as a sensing machine for an unmanned ground vehicle (UGV) is discussed in detail. The communication between the UGV and MAV is discussed and the planning of MAV missions to best serve the interest of UGV is presented.

1 INTRODUCTION

Autonomous machines and highly automated processes have an increasingly many operational sensing degrees of freedom: camera systems whose imaging areas can be chosen by focal length and pose, multi-sampling/multi-analysis chemical process analyzers, and web quality scanners are examples of operable sensing devices. With the development of Internet of Things machines are becoming capable of sharing sensing data online and thus they may request each other to carry out sensing missions for mutual benefits. In one extreme one of the autonomous machines is the master – e.g. a forest harvester or autonomous multipurpose loader – commanding a special purpose sensing machine – e.g. an unmanned aerial vehicle – to collect data relevant for the task allocated to the master machine [1,2]. In the other extreme the two machines are operating on equal terms each having its own task and performance criterion but benefitting from cooperative sensing and exchange of data. Optimal sensing is also strongly linked to the studies on perceptive behaviour of humans and animals [3,4].

Sensing adds value only if the information about the state of the system is uncertain. Thus the most natural framework for formulating and solving optimal sensing problems is the partially observable Markov decision processes (POMDP). If the system is given a task and performance measure in the form of maximizing the expected rewards (minimizing costs) during the task, sensing degrees of freedom appear in the problem on equal footing with regular control actions. Indeed, in the case of moving machines, sensing degrees of freedom are partly related to the move actions of the machine that are the regular control actions applied by the machine e.g. to find its way to a given location. A humanoid robot walking with some uncertainty and finding its way to a given goal location by using vision based simultaneous localization and mapping (SLAM) is an example of this: it seeks efficient policies of selecting walk lengths/directions after which observations for SLAM are made.

Completing a task under uncertainty should often be cautious: in addition to favoring plans that minimize the accumulated costs also the uncertainties about the costs are considered. In our topological route planning the uncertainties have been dealt with in two ways: either a plan is penalized by a term proportional to the variance of the costs [5], or the objective is to maximize the probability of staying below given costs [6]. Instead of carrying out a cost minimizing task a system may be only exploring its operational environment. In such a case the objective is to minimize the entropy of the state information. Offline solution methods of POMDPs for reward collecting cases – although computationally notoriously complex – can be approximated because the objective is a linear functional of the probability distribution of the state [7]. However, this no longer holds when maximizing the mutual information and hence online solutions based on tree searches must be applied [8,9].

This paper is organized as follows. Section 2 presents our robotic subsystems. Section 3 discusses the communication capabilities. Section 4 describes the key algorithmic development in our application.

2 ROBOTIC SYSTEMS AND SENSING SUBSYSTEMS

We have implemented an operable camera rig on the top of a commercially available indoors wheeled robot. The camera rig has three cameras the orientation of which can be changed on-line. Thus the system can be in three modes: three camera stereo-vision for accurate distance measurements, two-camera stereo-vision and a monocular vision in order to simultaneously get distance information to objects of interests and detection of previously undetected objects, and as three monocular cameras for covering robot's surroundings widely for object detection. The tasks for the robot are e.g. to navigate along a given route while avoiding collisions with dynamic objects in the scene. In order to manage this task, the robot must switch between the camera modes.

Nao is a commercially available humanoid robot used widely in man-machine interaction studies. Nao can be commanded to turn a given angle and then to walk a given distance. However, the Nao motion is rather uncertain, so that when commended to advance 1 m, the translational uncertainty is of the order of 10 cm and transverse uncertainty is of the order of 20 cm. Furthermore, Nao's vision-based localization during the walk is rather uncertain. For Nao to navigate in an environment of known obstacles, it must alternate between walking mode and vision-scanning mode. In the vision mode, Nao seeks markers that are detectable and then simultaneously localizes itself and maps the markers (SLAM). However, the robot must solve the problem: how long should the walks be in order for the robot to navigate to its target location in shortest time. In too short walks a lot of time is spent in SLAM, for too long walks, a lot of corrective actions are needed as the end point of the walk differs largely from the expected end point. Optimal cooperative sensing is studied with full scale machines. In the experimental setup an MAV (Micro Areal Vehicle) provides complementary abilities to an UGV (Unmanned Ground Vehicle). An MAV can provide information from a height and move fast, while it is not constraint to ground obstacles. An MAV equipped with appropriate sensory and communication capabilities is thus a strong candidate to act as a controllable sensor for a UGV on the ground.

UGV is a small multipurpose wheel loader capable of moving off-road with obstacles of 20cm high and 20deg ramps. It uses standard localization sensors: GNSS, wheel odometry, 3 axes gyros and 3 axes accelerometers, and EKF to generate 6D pose (3 positon and 3 orientation) estimates of the main body and the corresponding covariance matrix. Orientations are updated at the rate of 200 Hz while positions at 10 Hz. UGV is equipped with a tilting (nodding) 2D Laser scanner to provide a 3D range data. Appropriate transformation convert laser points given in the sensor frame to a global frame. Collection of all the range data point is thus called point cloud. Though nonlinear transformation, we use approximation and linearization to calculate uncertainties of the data in the global by including contributing uncertainties, namely those of pose estimation, calibration, and Laser sensor noise [2]. These calculations are done in real time xPC target module as shown in Figure 1.

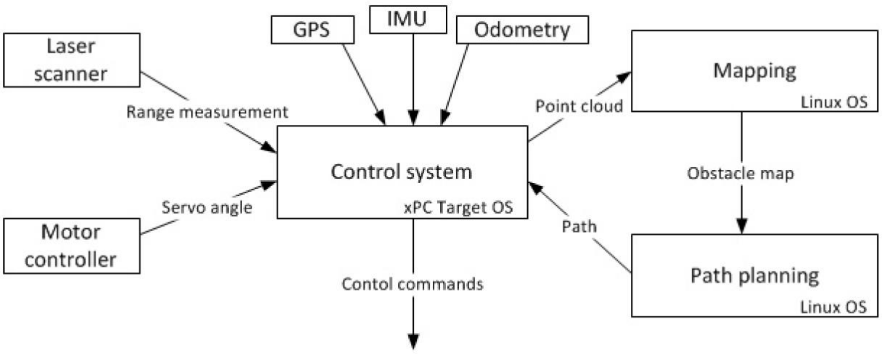

Figure 1. Navigation, mapping and planning architecture on UGV.

The MAV architecture is similar but with two main differences. First, for odometry, it uses visual odometry appropriate for aerial vehicle with limited computational power and down looking cameras. Second, MAV does not have a Laser scanner, but it uses ground looking stereo cameras and disparity of the images for point cloud generation. Both vehicles are capable of following trajectories and visiting ways points.

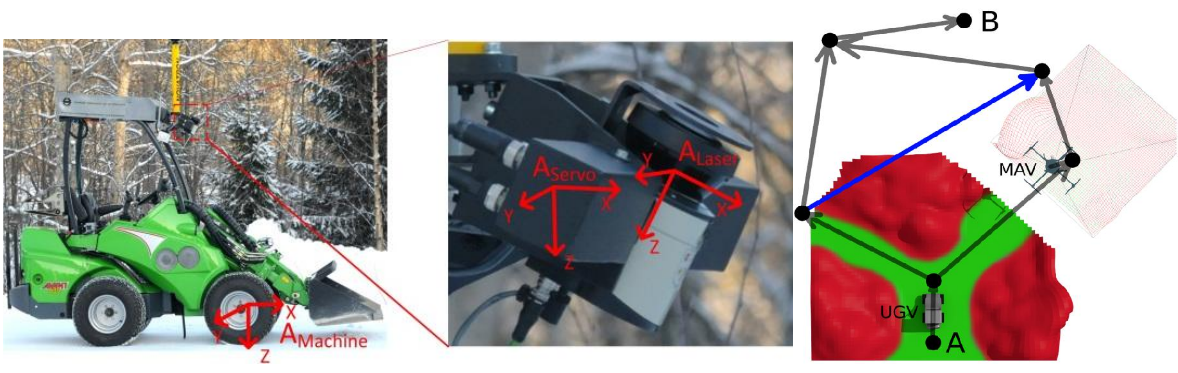

Figure 2. Left: UGV, multi-purpose wheel loader. Middle: tilting 2d laser scanner. Right: UGV and MAV cooperative sensing and planning scenario.

The environment is mapped on a 2D grid with each cell 20cm wide on both sides. Point clouds generated either on UGV or MAV are converted first to height maps. All points falling into one cell are assigned to the height of that cell by fusion based on Bayesian update. Data coming from later visits or the other vehicle are treated the same way. Thus the height map is a 2D matrix with mean and variance of the heights. This conversion is in a sense a resampling process. Based on the height map and gradient thresholding obstacle map is then generated. Ref. [2] provides more details on this and technics to improve accuracy of the obstacle map. The obstacle maps described above are metric maps. In addition to the obstacle map, we also use topological maps that define connectivity of the spatial locations of the same environments. Topological maps are best suited for high level path and mission planning, because they render the planning problem computationally tractable [10]. The topological map is modelled using graphs encoding traversal costs between nodes and their uncertainties.

Consider an UGV traveling from node A to node B in the topological map, as shown in Figure 2, right. Assume that some or all of the traversal costs of the graph edges are however initially highly uncertain. Therefore, optimal sensing planning is an instance of POMDP problem, where the planner must determine both UGV's traversal actions on the graph and possible execution of observation missions by the MAV. The cost to be minimized can be the length of the path, risk of taking a route (uncertainty about the traversability), or/and minus the chance of reaching a time deadline (for example in the case of MAV and limited battery power). Topological plans, which are sequences of nodes to be visited by MAV and UGV in a certain order, are then converted to equivalent metric locations on the obstacle map. A geometrical path is then planned that satisfies kinematic constraints of the UGV. Due to holonomic nature of the MAV metric way points are used directly.

Alternatively, the MAV may be used to reduce the uncertainty about the UGV's state. For example, in GPS denied area uncertainty of the state of the UGV will increase. MAV route can be planned to provide additional sensing data needed to reduce uncertainty.

3 COMMUNICATION BETWEEN MAV AND UGV

In practical cooperative sensing the communication between the machines is of vital importance. The system of a UGV, and an MAV described above is arranged as shown in Figure 3. In addition to the machines a user interface (UI) is available for tests.

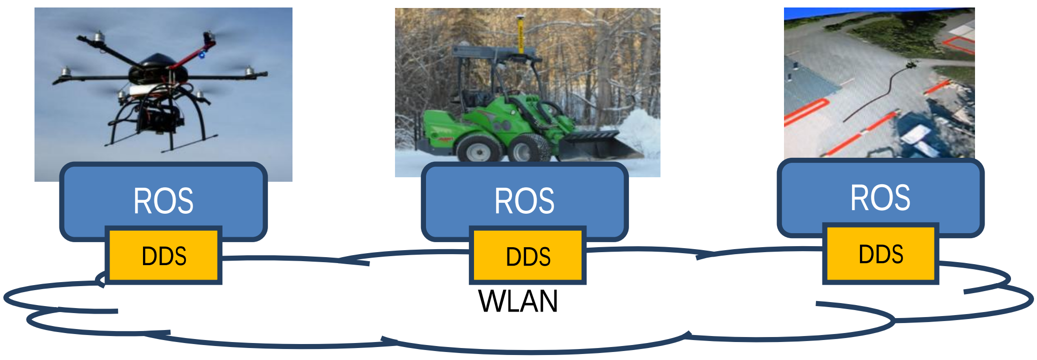

Figure 3. Communication architecture, MAV, UGV, UI.

For the UGV and MAV to cooperate, both must be able to execute their part of the task independently as single-vehicle operation, and to communicate relevant data and commands to each other. The main information to be shared is the obstacle map and the topological map for planning. The single-vehicle operation is handled by the on-board Robot Operating System (ROS) [11]. ROS is a robotic software framework based on publish/subscribe pattern that allows software nodes to communicate with each other. A message broker initializes the communication, after which the messages are sent directly between the subscriber and publisher nodes.

In robust multi-vehicle communication, ROS components are interfaced to a Data Distribution Service (DDS) communication middleware [12]. The MAV/UGV communication via WLAN is often unreliable. and can be lost occasionally. Although ROS can be used over a multi-vehicle network connection, if communication is lost between any ROS publisher, subscriber or message broker, it cannot be automatically renewed. DDS is an Object Management Group communication standard that enables scalable, real-time and high performance data distribution in a publish-subscribe manner. Quality of Service parameters are used to define the communication parameters such as required timeliness, liveliness and reliability of the data distribution. DDS is currently used e.g. by NASA for teleoperation and THALES for air traffic control. DDS is also planned to be used as communication method in major future versions of ROS, collectively denoted as ROS 2. To synchronize the messaged in the environment of each ROS core process having its own time the Network Time Protocol (NTP) is utilized. One of the computers will act as an NTP server while all the other are clients.

Stereo image processing to produce point clouds is computationally heavy. Due to limitations of computational power and communication due to our WLAN, the computational load needs to be distributed between MAV and UGV. The bandwidth limitations prevent distribution of raw images. Thus the sensor data is pre-processed into 3D occupancy grids before sending it to UGV. The common obstacle map generation and optimal observation planning algorithms run on the UGV. This distribution of tasks results in manageable data flow in the network.

4 ALGORITHMS FOR OPTIMAL PLANNING OF SENSING

Optimal planning for sensing is a POMDP problem that has been widely studied, see [7] , and thus solved with dynamic programming. The objective is to maximize the prior expectation of posterior optimal performance: one must consider each possible sensing data that may result from a sensing action, find out which action towards completing the task would be taken as a result of that data, assess the resulting performance, construct the probability distribution of performances resulting from a sensing action taking into account the prior distribution of data values, and finally choose the sensing action that has the most attractive distribution of the performances.
Optimal sensing can be solved in offline or online. Offline solution provides the optimal control action as a lookup for any current state information whereas online solution is produced each time for the current state information only. Offline solutions are extremely hard to compute but results being lookups, they are quick to use. Online solutions are usually open loop in that they find the optimal sequence of actions without considering how the future actions depend on the data between current time and the actuation time. Closed loop solutions provide the optimal actions as functions of the future data collected before the actuation time. The performance of the open loop solutions is usually improved by the receding horizon principle.

We have considered the algorithms for optimal sensing for UGV/MAV cooperation in topological-map-based exploitation when the attitude towards risks is taken into account [6,7], and in exploration for the wheeled robot [8,9]. We solved the Nao walk length problem with particle filtering SLAM and QMDP planning.

5 CONCLUSIONS

This paper provides an overview on our research environment for optimal sensing and the results achieved. When implementing practical systems with optimal operation of sensing, a robotic platform, sensing degrees of freedom via operable sensors on a robot or special purpose sensing machines, communication between the machines, and algorithms for state estimation and planning optimization are needed. The paper presented some approaches to meet the requirements of optimal cooperative sensing, but the methods and techniques need to be adapted to the specifics of the problem rather than expecting universally successful approaches.

Acknowledgments. The authors RG, KH, MH, JM, JPR, AR and RR express their gratitude to support by the Academy of Finland, grant “Optimal operation of observation systems in autonomous mobile machines” (O3-SAM).

6 REFERENCES

[1] J. Melin, M. Lauri, A. Kolu, J. Koljonen, R. Ritala, ”Cooperative Sensing and Path Planning in a Multi-Vehicle Environment”, submitted to IFAC Workshop on Advanced Control and Navigation for Autonomous Aerospace Vehicles (2015).

[2] A. Kolu, M. Lauri, M. Hyvönen, R. Ghabcheloo, "A mapping method tolerant to calibration and localization errors based on tilting 2D laser scanner," European Control Conference, 2015, accepted for publication

[3] A. Clark, “Whatever next? Predictive brains, situated agents, and the future of cognitive science”, Behavioral and Brain Sciences 1-73, (2013) and references therein.

[4] K. Friston, “Learning and inference in brain”, Neural Networks 16(9): 1325-1352 (2003).

[5] A. Ropponen, M. Lauri, R.Ritala, “Risk-averse shortest paths with observation options and Gaussian costs”, submitted to Autonomous Agents and Multi-agent Systems.

[6] M. Lauri, A. Ropponen, R. Ritala, ”Meeting a deadline: shortest paths on stochastic acyclic graphs with allocation of observation resources”, submitted to European Journal of Operational Research.

[7] see S. Thrun, W. Burgard, D. Fox “Probabilistic Robotics), Chs 15-17, (MIT press, 2006).

[8] M. Lauri, R. Ritala “Stochastic control for maximizing mutual information in active sensing”, in IEEE International Conference on Robotics and Automation, Workshop Robots in Home and Industry: Where to Look First (2014).

[9] M. Lauri, R. Ritala "Optimal Sensing via Multi-armed Bandit Relaxations in Mixed Observability Domains", accepted to ICRA 2015.

[10] Konolige, K. ; Marder-Eppstein, E. ; Marthi, B., “Navigation in hybrid metric-topological maps,” IEEE International Conference on Robotics and Automation (ICRA), 9-13 May 2011

[11] Quigley, M., Conley, K., Gerkey, B., Faust, J., Foote, T.,Leibs, J., Wheeler, R., and Ng, A.Y. (2009). ROS: an open-source Robot Operating System. In ICRA workshop on open source software.

[12] Object Management Group (2007). Data distribution service for real-time systems (dds), version 1.2, formal/2007-01-01, january 2007. http://www.omg.org/spec/DDS/1.2. Accessed: 2014-12-12.