Sensor Data Fusion for Mobile Robot

Věchet Stanislav*, Krejsa Jiří*
*Brno University of Technology, Faculty of Mechanical Engineering, Brno, Czech Republic,
e-mail: *vechet.s@fme.vutbr.cz krejsa@fme.vutbr.cz*

***Abstract*— Autonomous mobile robot must be equipped with a number of sensors of various measurement principles. The data fusion of measured data is essential for successful navigation of the robot. The paper describes the data fusion method based on Bayesian network. Apart from theoretical grounds of the used approach, the example is also given fusing the compass, GPS and odometry sensor data, because such sensors are commonly present in outdoor robots.**

***Keywords*— Automotive Application, Robotics, Sensor.**

I. Introduction

The understanding of basic principles in data fusion is the key knowledge that must be gained by the students of robotics regardless their later specialization. Bayesian filters are often difficult for the students to handle without prior understanding of underlying statistics. Bayesian networks can help the students to cope with the principles while at the same time the gained knowledge can be used directly as a tool in data fusion. The paper gives the detailed description of such a case.

Sensors data fusion belongs to one of the essential issues in mobile robotics. When the sensor suite of a mobile robot includes several sensors of different types the data fusion is necessary. Combining the sensor readings, the robot is designed to accomplish various tasks such as constructing a map of its environment, localizating itself in given map or recognizing objects that should be avoided [1]

There are several different aproaches, how data fusion methods are designed and used.

The data fusion is often used, because of its robustnes, for calculating the position and orientation of an autonomous mobile robot[3], as the fusion system is distributed, robust, and asynchronous. It is robust because the system is designed to keep working properly in spite of the failure, removal, or change in sensors configuration. The implementation of the reliable fusion system is based on distributed version of the popular Kalman filter developed by Durrant-Whyte and Rao.

Data fusion can generally be divided into the three main groups which can be classified according to sensors configuration as follows:

competitive - different types of sensors are used to measure the same attributes of given environment; usually there is information redundancy, which could be a source of errors,

complementary - each sensor reads different attribute of the same environment, this could be and disadvantage in case of sensor failure

cooperative - in that case one sensor depends on the other; they have to work together.

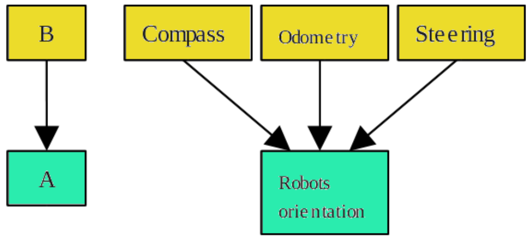

Fig. 1. The simplest Bayesian network (left) and the practical example (right).

Presented paper deals with a method which was succesfuly tested for complementary data fusion on mobile robot for outdoor environment equiped with several sensors of different nature. In particular, the experimental robot carries a popular laser range finder SICK, ultrasonic and infrared distance sensors, compass, GPS recever, digital camera and odometry IRC sensors. As the sensors are of variant sensing principle and purpose, the basics idea is to provide the proper information regarding robots position and internal states. In such a case the data fusion is a key algorithm.

So far we have been using complementary data fusion in all our robots. For such a purpose we have been developing a method of data fusion based on Bayes theorem, which is the base for most probabilistic method in robotics.

Presented paper describes the basic method for fusion of data acquired via odometry, compass and steering angle. Naturally, its easy to widespread this basic set of used sensors with other sensing devices. Therefore this method can be used as a main tool for measured data fusion.

Finally, two different environments were prepared to test the method with the real robot and the results from those experiments are discussed in chapter V.

II. Bayes Theorem

Bayes theorem belongs to a basic methods to deal with conditional probability. More precisely it relates the conditional probability of events A and B. Its well known how to derive it from basic conditional probabilities equations (for example in [4]), so here just the final state of the Bayes theorem equation is presented:

$$P(X|Y) = \frac{P(Y|X)}{P(Y)} \quad (1)$$

where:

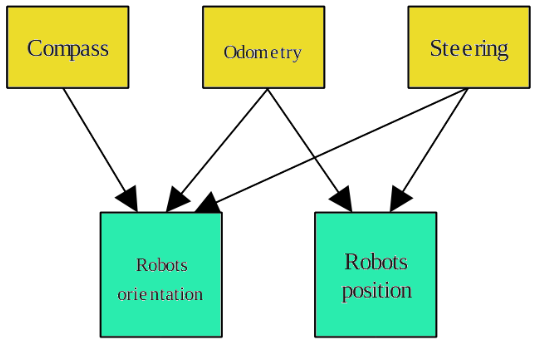

Fig. 2. Data fusion via Bayesian network.

- $P(X)$ is the prior probability of X; in sense what we know about X at the beginning, it is independent on any others variables,
- $P(X|Y)$ is called posterior probability; the conditional probability of X given$Y$, usually this probability represents the information what we are interested for: what is the probability of beeing in given state X if the robot sensors measured data Y.
- $P(Y|X)$ is the conditional probability of Y given X, sometimes called also the invers probability beacause it represents the situation: what is the probability that the collected measurement Y was measured in given state X.
- $P(Y)$ is the prior probability of Y; it acts as normalizing constant and if we have complete sets of possible states and measurements, this could be calculated based on total probability law.

As its shown, the main idea is based on $P(X|Y)$ calculation, if we know so-called inverse probability $P(Y|X)$. This probability can be obtained as the invers model of solved problem or by measurements on real system with collected input and output information. Afterwards, the paired set of inputs and related outputs is used for probability calculations.

Bayes theorem can be easily used in more complicated relationships with more then two events. One of possible applications of the Bayes Theorem is in Bayesian networks [6] or filters.

The primitive relation of two events described above could be considered the simplest network (see figure 1 - left). Basically, the Bayesian networks are primarily used for more complex relationship description (see figure 1 - right). On that figure the relations between some of the sensors discussed above are shown. Those relations are naturally created as it is intuitive for human. Its easy to derive how the final orientation of the robot is influenced by the sensors. Moreover, the existing Bayesian network can be easily modified to solve a more complicated problem (see figure 2).

III. Bayesian Network Simulator

There are three different sources of information which possibly could be used to calculate the robots true orientation (the heading angle of the robot):

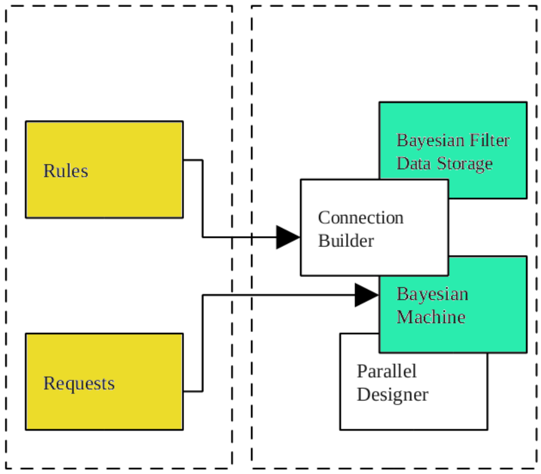

Fig. 3. Bayesian network simulator internal structure.

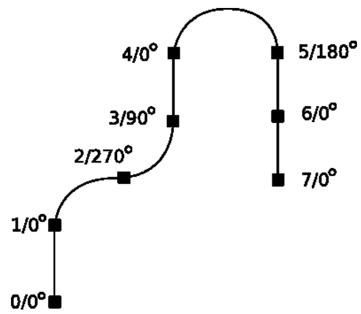

Fig. 4. Path followed by autonomous robot; Position/Robot orientation relative to robots previous orientation.

- the compass is used for absolute orientation measurement. Information from this kind of sensor can be easily converted to robots orientation,
- odometry give the indirect information about a relative change of the orientation, which has to be calculated from traveled distances of left and righ wheels,
- steering angle give also the indirect information about the relative changes in robots orientation. The change has to be calculated from the driving model (ackerman, differential, etc.).

To obtain the resulting orientation of the robot, the additional calculations performing the data fusion are necessary. Usually the data fusion from such different information sources is not straighforward.

We have prepared the simulator of Bayesian network to be able to work efficiently under various conditionas. The simulator internal structure is shown on figure 3.

The implementation of Bayesian network simulator consists of six important blocks:

Rules - user definition of problem structure, it means that the user is able to define given problem by set of high-level commands. These commands are based on written rules for probabilistic equations.

Requests - user requests to the simulator, represented by a set of possible commands, which define the requests to the Bayesian machine.

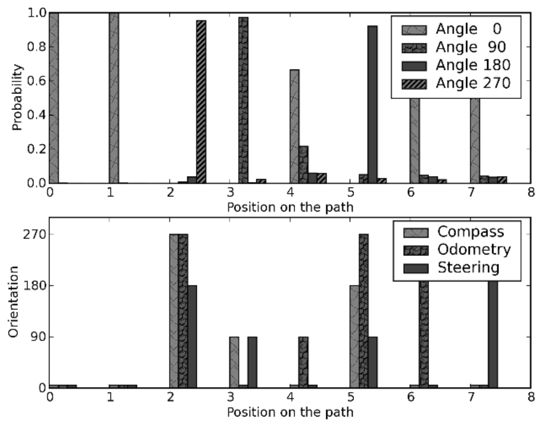

Fig. 5. Robot orientation identification via Bayesian network.

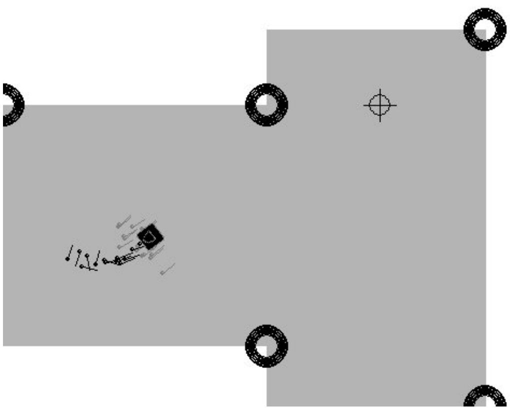

Fig. 6. Initial part of the path traveled by the robot in the Map-I.

Connection Builder - interface between user inputs definition and the real data structure, the probabilistic rules are translated into the form to be understood by the Bayesian machine.

Bayesian Filter Data Storage - the main data strorage, it holds the complete structure of given problem, probabilities and possible results.

Bayesian Machine - acts as a main computation framework, it computes all necessary probabilities requested by the user, there is an automatic inference mechanism based on Bayesian network theory as well as implementation of the set of mathematical equations and necessary probabilistic laws.

Parallel Designer - Bayesian network can be implemented as a parallel algorithm so this block organizes the parallel operations.

Experimental results obtained from presented simulator are shown in followed chapter as we have used it as a main tool for data fusion of robot orientation measurements.

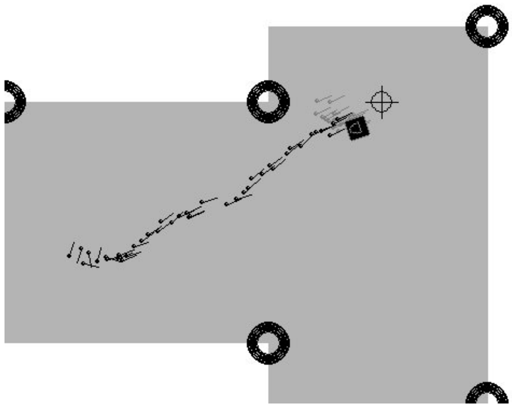

Fig. 7. Complete path traveled by the robot.

IV. Data Fusion via Bayesian Network

To detect that our method to data fusion works properly, a simple simulation experiment with autonomous robot was first prepared. The robot was equipped with three means to determine its relative orientation on followed path. First one was thecompass which was used to measure absolute orientation of the robot. The second one was an odometry reading and the change in orientation was calculated from the difference in traveled distances of each single wheel. The orientation of the robot calculated from the steering angle represents the third variable.

The path traveled by the robot is shown on figure 4. There are seven number of checkpoints in which the orientation was measured. On that figure one can see the position number/true robot orientation in degrees relative to the robot previous location. The orientation was measured by all three methods and we use previously described tool to fuse the data.

The simulation results are shown on figure 5. On the bottom graph the orientation measured by compass, odometry and steereing in each position on the path is shown. The upper graph shows the probabilities that the robot is oriented 0, 90, 180 or 270 degrees. One can see that the highest probability (the output of the Bayesian network) in each point corresponds with the true orientation of the robot (see figure 4).

The best example illustrating how the data fusion works can be seen in point 5 on figure 5. The robots real orientation is 180 degrees and the compass measured that orientation properly. On the other hand the orientations calculated from odometry and steering were wrong (odometry 270 degrees, steering 90 degrees), but the probability that the robots orientation is 180 degree is still the highest. This is caused by different probabilities of correct orientation measurement.

Note, that compass measures with the same probability in all directions independently $P(Compass) = 0.95$. However, odometry and steering have high probabilities of success for angle 0, for example $P(Odometry = 0) = 0.95$, less for angle 90 or 270 $P(Odometry = 90) = 0.5$ and even lower probability for angle 180 $P(Odometry = 180) = 0.3$. These probabilities were measured in various practical experiments and they causes that if compass measured angle 180 degrees and odometry and steering 90 or 270 degrees, than these two values have less influence to the calculated orientation.

Presented simple example illustrates how to use Bayesian network in basic task. The method can be easily extended to be used in more challenging problems including continuous variables [4]. In such a case the discretization of corresponding variable is necessary.

V. Real experiments

The method described above was tested in various environments to ensure that the method is capable of working under the real conditions. The real experiments were prepared as follows.

Two different environments of different dimensions were prepared. The smaller one has the size of approximately 5x5 meters (see figures 6 to 10), while the bigger one is about 8x15 meters (see figures 11 to 13).

A. Map I

The robots goal was to reach the target position while the map was initially known and it performs the localization procedure during the travel. Only the orientation variable out of the localization estimate is depicted as it was fused by the Bayesian network. The xy coordinates were obtained by different localization method (not presented in this paper).

Each figure shows different part of the way of the robot. The real position of the robot in actual step is marked with black rectangle and the target position is marked by black circle.

Localized positions and fused orientations through the whole path traveled in the environment is marked by points with short line as the robots heading angle direction.

Figures 6 to 7 shows different stages of the path traveled by the robot through the given map. As the reader can clearly see, fused orientations are in corelations with the true orientation.

The complete path traveled by the robot is shown on figure 8. There can be seen that the estimated path is close to the true path of the real robot.

While the path traveled by the real robot is smooth (see figure 8 - real position), the estimated path is more discontinuous. This is caused by the precision of the probabilistic estimator. The error in position estimation is shown on figure 9.

Figure 10 shows the difference (in degrees) between the real and the fused orientation of the robot obtained via Bayesian network. The error in heading angle estimate is caused by nonhomogenous magnetic field and tires slip, however we found the results reasonable.

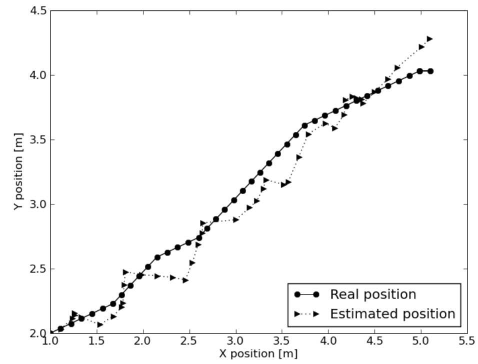

Fig. 8. Real and estimated path of the robot.

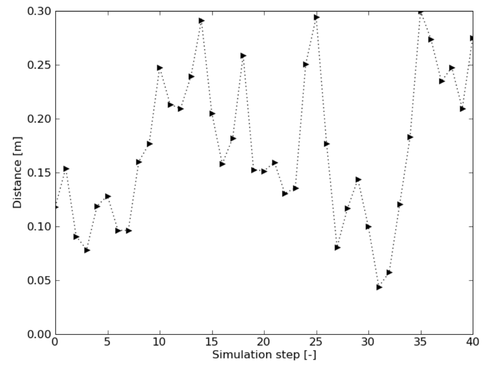

Fig. 9. Diferences between estimated and measured position.

B. Map II

Another test was performed in larger area covered about 100 square meteres. The task the robot should perform is to travel to the target position about 12 meters away.

The complete path with the dimensions of the experimental area is shown on figure 11. The estimated path folows the real path traveled by the robot and the precision is in same range as in experiments performed in Map I and is shown o figure 12 (compare with figure 9).

The comparison of real orientation of the robot and the fused orientation obtained via Bayesian network is shown on figure 13. The precision is also similar to the smaller map presented in previous chapter.

VI. Conclusion

We have presented the basic simulation and real experiments with data fusion via Bayesian network. Three different sensors (digital compass, odometry and steering) were used as the information source to estimate the orientation (heading angle) of the mobile robot and the fusion method based on Bayesian network was successfully applied.

The key issue in the method itself is proper determination of condition probabilities of mutually related events. The simplest approach to determine the probabilities is to directly use known errors in sensor measurement, or to create an inverse model of the problem.

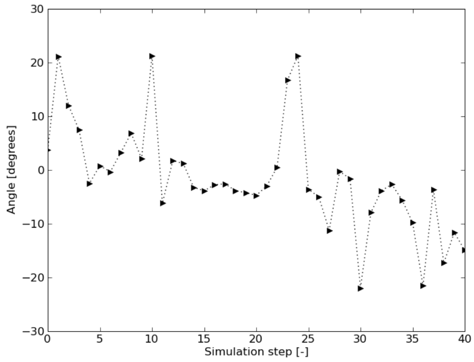

Fig. 10. Difference between the real and estimated orientation.

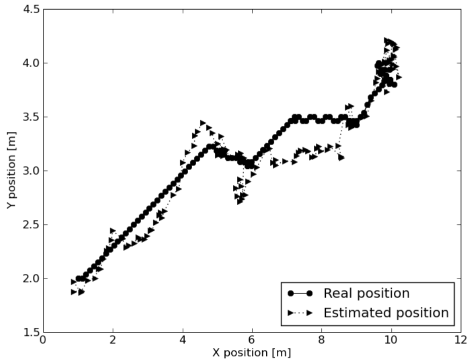

Fig. 11. Comparison of the real and estimated path of the robot in the Map II.

Bayesian networks are simple yet powerful tool in data fusion field. Understanding the Bayesian theorem that is the base for probabilistic approach so popular nowadays in mobile robotics is essential when more complex methods and algorithms are applied.

We found described examples as both directly usable and of high educational value as the problem of heading angle determination from several sensors is for students easy to understand and thus uncover the underlying principles of probabilistic robotics.

Acknowledgment

Published results were acquired with the support of the Brno University of Technology, research plan FSI-S-10-29 "The development of sensors and control of unconventional mechatronic devices".

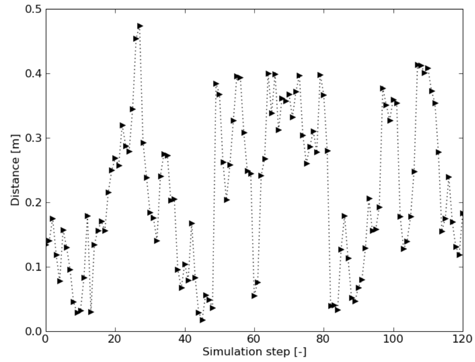

Fig. 12. The diference between real angle and fused angle via bayesian network, Map II.

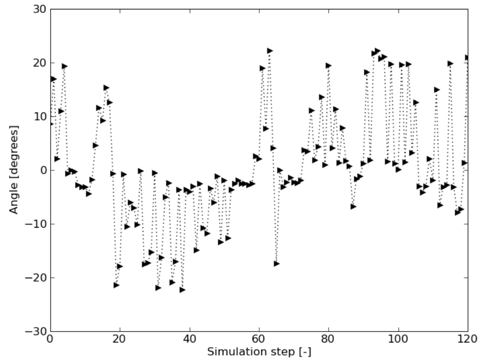

Fig. 13. The diference between real angle and fused angle via bayesian network, Map II.

References

[1] M. Kam, Z. Xiaoxun, P. Kalata, "Sensor fusion for mobile robots navigation," *Proceedings of the IEEE, Data Fusion lab., Drexel univ., Philadelphia.,* pp.108-119, 1997.

[2] Jin, Tae-Seok and Lee, Jang Myung and Tso, S. K., "A new approach using sensor data fusion for mobile robot navigation," *Robotica,* volume 22, pp. 52-59, 2004.

[3] J. A. Lpez-Orozco, J. M. de la Cruz, E. Besada, "An Asynchronous, Robust, and Distributed Multisensor Fusion System for Mobile Robots," *The International Journal of Robotics Research,* Vol. 19, No. 10, 914-932 (2000).

[4] A. Singhal, Ch. Brown, "Dynamic Bayes Net Approach to Multimodal Sensor Fusion," *Proceedings of the SPIE - The International Society for Optical Engineering,* pp.2-10, 1997.

[5] F. V. Jensen, "Bayesian networks and Decision Graphs," *Statistics for Engineering and Information Science,* Springer, Berlin, Heidelberg, 2001.

[6] K. P. Murphy, "Dynamic Bayesian Networks: Representation, Inference and Learning," *PhDthesis,* UC Berkley, Computer Science Division, 2002.

[7] F. V. Jensen, "An Introduction to Bayesian Networks," *UCL Press,* 1996.