Proceedings 2003 IEEE International Symposium on
Computational Intelligence in Robotics and Automation
July 16-20, 2003, Kobe, Japan

Path Evaluation for a Mobile Robot Based on a Risk of Collision

Masahiro Irie
Graduate School of Natural Science and Technology,
Okayama University

Keiji Nagatani
Graduate School of Natural Science and Technology,
Okayama University
http://usm.sys.okayama-u.ac.jp/~keiji

Akio Gofuku
System Engineering Dept.
Okayama University

Abstract

An odometry system that mobile robot uses for positioning has cumulative error because of wheels' slippage and uneven ground. It causes a risk of collision to obstacles. Therefore, we propose a path evaluation method for a mobile robot based on a risk of collision.

To evaluate a robot's path, we define an evaluation value as an integral of a risk of collision along the path. To evaluate the risk of collision at each point, we use an estimated positioning error generated in the odometry system. Using the evaluation method, the robot can plan a path based on a risk of collision, not the shortest path. We also consider sensing points planning for position adjustment of the mobile robot, based on the same approach. Some examples of path evaluation results support a validity of the proposed method.

1 Introduction

This research aims to construct a path evaluation method for navigation of a mobile robot based on a risk of collision. Many types of wheeled mobile robots use an odometry system to estimate its position and orientation. However, the odometry system has a cumulative positioning error problem because of wheels' slippage, uneven ground and errors of robot's model. It causes a risk of collision to obstacles. To reduce the risk, the robot should choose safer path before starting navigation. Therefore, to evaluate robot's path is an important for mobile robot's navigation.

To evaluate a risk of collision, we review a method to express the positioning error (or uncertainty) by an error ellipse. Then, we define a risk of collision as the inverse of Mahalanobis distance of the error ellipse that contacts obstacles. In other words, a risk of collision is increased while its error ellipse is expanded.

On the other hand, to reduce a positioning error caused by an odometry system, usually external sensor is used to adjust the robot's position by detecting known landmarks. Therefore, we assume that a risk of collision can be reduced by sensing landmarks.

On the basis of above features of a risk of collision, we propose a path evaluation method as an integral of a risk of collision along a path. To evaluate the risk of collision at each point, we use a positioning error generated in the odometry system.

In this paper, we introduce a method of estimating positioning error of a mobile robot using an odometry system. Then we propose a method to evaluate paths based on a risk of collision. Finally we show two simulation examples of path evaluation with a use of an external sensor.

2 Previous Works

Usually, path evaluation is discussed together with algorithms of path planning. A variety of algorithms for path planning of mobile robots are summarized in [5]. In many of these approaches, safe path for mobile robot is considered as further path from obstacles. However, we have an experience that "further path" is not enough solution for safety navigation in real environment.

To improve it, "path planning method based on a risk of collision" was proposed in [7]. This approach uses an evaluation function to minimize a risk of collision, related with a positioning error of the mobile robot and sensing costs. However, the positioning error was assumed to be increased monotonously according to robot's motion, and positional relationship between a robot and obstacles was not considered.

From the point of view of robot's positioning, there were several researches that uses positional uncertainty for a mobile robot by an odometry system. Theoretical discussion of expression of the uncertainty was arranged in [6]. Moon et al. [3] and Kosaka et al. [1] proposed a method of "vision based navigation" for a mobile robot with considering uncertainty of an odometry system. Komoriya et al. proposed an acquisition method of the most appropriate landmark, with considering odometry error [4]. Roy et al. proposed "coastal navigation approach". It minimizes the likelihood of getting lost by navigating neighbor area of the map that have high information content [8].

In above approaches, a risk of collision is considered while robot is navigated. However, those are not considered at the stage of path planning. In our approach, we evaluate some path candidates to select the optimal one in path planning stage, with estimating positioning error of the robot and considering positional relationship between a robot and obstacles.

3 Criteria and Assumptions for Path Evaluation

We assume the following conditions.

- Our target robot is a wheeled mobile robot that has an odometry system that estimates robot's position and orientation.
- A positioning error in the odometry system can be reduced by detecting landmarks using external sensors. (E.g. vision sensor.)
- The robot knows a start point, a goal point and static environment model that includes obstacles and sensing points.
- The shape of the robot is assumed to be circle for path evaluation.

0-7803-7866-0/03/$17.00 ©2003 IEEE

In above assumptions, we assume that the robot already knows some path candidates from a start point to a goal. Then our path evaluation method selects a better path for the robot's navigation from the point of view of a risk of collision.

In our experience, "further path from obstacles" is not enough for a path evaluation based on safety. Therefore, we consider that the path that keeps "low risk of collision" is better for a robot's navigation. We define that a path is evaluated by integrating a risk of collision from the start point to the goal. To evaluate the risk of collision at each point, we use a positioning error generated in the odometry system. In this paper, we propose a path evaluation method based on above criteria.

4 Position Estimation and Positioning Error in an Odometry System

In this section, we introduce a method that expresses uncertainty of position in an odometry system. Basically we regard a covariance matrix (that defines an error ellipse) as uncertainty of robot's position, that was discussed in [6]. A model of the target robot is shown in Fig. 1.

4.1 Covariance Matrix of Positioning Error

When a position and orientation of the robot are expressed by $(x[t], y[t], \theta[t])$, and velocity and angular velocity are expressed by $(v[t], \omega[t])$, position and velocity vectors are expressed by following equation,

$$\mathbf{P}[t] = \begin{bmatrix} x[t] & y[t] & \theta[t] \end{bmatrix}^T \quad (1)$$

$$\mathbf{V}[t] = \begin{bmatrix} v[t] & \omega[t] \end{bmatrix}^T \quad (2)$$

where T means transposition.

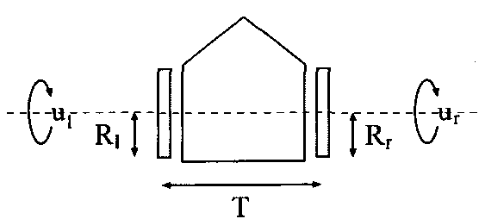

Figure 1: Model of a robot with power wheeled steering

Then the position after sampling time τ[sec] is expressed by

$$\mathbf{P}[t+\tau] = \mathbf{P}[t] + \tau \begin{pmatrix} v[t]cos(\theta[t]) \\ v[t]sin(\theta[t]) \\ \omega[t] \end{pmatrix} + \tau\mathbf{n}[t] \quad (3)$$

where $\mathbf{n}[t]$ is the error caused by calculation error and other errors.

Now $f(\mathbf{P}[t], \mathbf{V}[t])$ is expressed as the right side of equation (3) expect $\tau\mathbf{n}[t]$, $\hat{\mathbf{P}}[t]$ is an estimated position of $\mathbf{P}[t]$, $\Delta\mathbf{P}[t]$ is the error of it, $\hat{\mathbf{V}}[t]$ is the observed velocity of $\mathbf{V}[t]$ and $\Delta\mathbf{V}[t]$ is the error of it. In this case, $\Delta\mathbf{P}[t], \Delta\mathbf{V}[t]$ and $\mathbf{n}[t]$ (expectation values) are small enough.

Using above expression, $\mathbf{P}[t+\tau]$ is regarded as following equation by using second order of Tayler expansion.

$$\begin{aligned} \mathbf{P}[t+\tau] &= f(\mathbf{P}[t], \mathbf{V}[t]) + \tau\mathbf{n}[t] && (4) \\ &= f(\hat{\mathbf{P}}[t] + \Delta\mathbf{P}[t], \hat{\mathbf{V}}[t] + \Delta\mathbf{V}[t]) + \tau\mathbf{n}[t] && (5) \\ &\simeq f(\hat{\mathbf{P}}[t], \hat{\mathbf{V}}[t]) + \mathbf{j}[t]\Delta\mathbf{V}[t] + \mathbf{k}[t]\Delta\mathbf{V}[t] + \tau\mathbf{n}[t] && (6) \\ &= \hat{\mathbf{P}}[t+\tau] + \Delta\mathbf{P}[t+\tau] && (7) \end{aligned}$$

where $\mathbf{j}[t]$ and $\mathbf{k}[t]$ are Jacobian matrix as follows.

$$\mathbf{j}[t] = \left.\frac{\partial f(\mathbf{P}, \mathbf{V})}{\partial \mathbf{P}}\right|_{\hat{\mathbf{P}}[t], \hat{\mathbf{V}}[t]} \quad (8)$$

$$\mathbf{k}[t] = \left.\frac{\partial f(\mathbf{P}, \mathbf{V})}{\partial \mathbf{V}}\right|_{\hat{\mathbf{P}}[t], \hat{\mathbf{V}}[t]} . \quad (9)$$

In our case, a robot with "power wheeled steering mechanism" (in Fig.1) is assumed as a target robot. Therefore (8) and (9) are expanded as following.

$$\mathbf{j}[t] = \begin{pmatrix} 1 & 0 & -\tau\hat{v}[t]sin(\hat{\theta}[t]) \\ 0 & 1 & \tau\hat{v}[t]cos(\hat{\theta}[t]) \\ 0 & 0 & 1 \end{pmatrix} \quad (10)$$

$$\mathbf{k}[t] = \begin{pmatrix} \tau\hat{v}[t]cos(\hat{\theta}[t]) & 0 \\ \tau\hat{v}[t]sin(\hat{\theta}[t]) & 0 \\ 0 & \tau \end{pmatrix} . \quad (11)$$

According to the fact of $\hat{\mathbf{P}}[t+\tau] = f(\hat{\mathbf{P}}[t], \hat{\mathbf{V}}[t])$ in (6) and (7), updated equation of the error $\Delta\mathbf{P}[t+\tau]$ is expresses as

$$\Delta\mathbf{P}[t+\tau] = \mathbf{j}[t]\Delta\mathbf{P}[t] + \mathbf{k}[t]\Delta\mathbf{V}[t] + \tau\mathbf{n}[t]. \quad (12)$$

where $\Delta\mathbf{P}[t], \Delta\mathbf{V}[t]$ and $\mathbf{n}[t]$ are non-correlated each other.

On the other hand, covariance matrix $\Sigma_{\mathbf{P}}[t]$ is expressed as

$$\begin{aligned} \Sigma_{\mathbf{P}}[t] &= E(\Delta\mathbf{P}[t]\Delta\mathbf{P}[t]^T) && (13) \\ &= \begin{pmatrix} \sigma_x[t]^2 & \sigma_{xy}[t] & \sigma_{x\theta}[t] \\ \sigma_{xy}[t] & \sigma_y[t]^2 & \sigma_{y\theta}[t] \\ \sigma_{x\theta}[t] & \sigma_{y\theta}[t] & \sigma_\theta[t]^2 \end{pmatrix} . && (14) \end{aligned}$$

Therefore, updated equation of the covariance matrix $\Sigma_{\mathbf{P}}[t]$ per sampling time τ[sec] is,

$$\Sigma_{\mathbf{P}}[t+\tau] = \mathbf{j}\Sigma_{\mathbf{P}}[t]\mathbf{j}^T + \mathbf{k}\Sigma_{\mathbf{V}}[t]\mathbf{k}^T + \tau^2\Sigma_{\mathbf{n}}[t] \quad (15)$$

where $\Sigma_{\mathbf{V}}[t]$ and $\Sigma_{\mathbf{n}}[t]$ are expressed as follows.

$$\Sigma_{\mathbf{V}}[t] = E(\Delta\mathbf{V}[t]\Delta\mathbf{V}[t]^T) \quad (16)$$

$$\Sigma_{\mathbf{n}}[t] = E(\mathbf{n}[t]\mathbf{n}[t]^T). \quad (17)$$

4.2 Covariance Matrix of Velocity Error

To calculate covariance matrix $\Sigma_{\mathbf{P}}[t+\tau]$ in (15), we should figure out a covariance matrix $\Sigma_{\mathbf{V}}[t]$.

In Fig.1, estimated rotational speeds of wheels are expressed by $\hat{u}_l[t]$ and $\hat{u}_r[t]$, the estimated radii of wheels are $\hat{R}_l$ and $\hat{R}_r$, and the estimated tread is $\hat{T}$. Then an estimated value of velocity ($\hat{v}_0[t]$) and the angular velocity $\hat{\omega_0}[t]$ are expressed by following equation.

$$\hat{v}_0[t] = \frac{\hat{R}_r\hat{u}_r[t] + \hat{R}_l\hat{u}_l[t]}{2} \quad (18)$$

$$\hat{\omega}_0[t] = \frac{\hat{R}_r\hat{u}_r[t] - \hat{R}_l\hat{u}_l[t]}{\hat{T}}. \quad (19)$$

In this research, we assume that the errors of rotational speed are ignored, because the encoder is very accurate.

Then, we define $\hat{\mathbf{m}}$ as the estimated vector of wheel radii and tread, and $\Delta\mathbf{m}$ as the error of $\hat{\mathbf{m}}$.

$$\hat{\mathbf{m}} = \left[\begin{array}{ccc} \hat{R}_l & \hat{R}_r & \hat{T} \end{array}\right]^T \quad (20)$$

$$\Delta\mathbf{m} = \left[\begin{array}{ccc} \Delta R_l & \Delta R_r & \Delta T \end{array}\right]^T. \quad (21)$$

Then, the covariance matrix of velocity error $\mathbf{\Sigma_V}[t]$ is expressed as follows.

$$\begin{aligned} \mathbf{\Sigma_V}[t] &= \begin{pmatrix} \sigma_{v_0}[t]^2 & \sigma_{v_0\omega_0}[t] \\ \sigma_{v_0\omega_0}[t] & \sigma_{\omega_0}[t]^2 \end{pmatrix} & (22) \\ &= \mathbf{L}[t]\mathbf{\Sigma_m}\mathbf{L}[t]^T & (23) \end{aligned}$$

where $\mathbf{\Sigma_m}$ is a covariance matrix of wheel rotational speeds, wheel radii and tread error. These are non-correlated, therefore it is expressed as

$$\begin{aligned} \mathbf{\Sigma_m} &= E(\Delta\mathbf{m}\Delta\mathbf{m}^T) & (24) \\ &= \begin{pmatrix} \sigma_{R_l}^2 & 0 & 0 \\ 0 & \sigma_{R_r}^2 & 0 \\ 0 & 0 & \sigma_T^2 \end{pmatrix} & (25) \end{aligned}$$

where $\mathbf{L}[t]$ is Jacobian matrix as follows.

$$\begin{aligned} \mathbf{L}[t] &= \left.\frac{\partial\mathbf{V}[\mathbf{m}]}{\partial\mathbf{m}}\right|_{\hat{\mathbf{m}}} & (26) \\ &= \begin{pmatrix} \frac{\hat{u}_l[t]}{2} & \frac{\hat{u}_r[t]}{2} & 0 \\ -\frac{\hat{u}_l[t]}{\hat{T}} & \frac{\hat{u}_r[t]}{\hat{T}} & -\frac{\hat{R}_r\hat{u}_l[t]-\hat{R}_l\hat{u}_l[t]}{\hat{T}^2} \end{pmatrix} & (27) \end{aligned}$$

Once we assume each variance in the equation (25), we can calculate a covariance matrix $\mathbf{\Sigma_V}[t]$ in (22), and the covariance matrix of position uncertainty can be updated using the equation (15).

4.3 Error Ellipse

Usually, a robot's position uncertainty is expressed by an error ellipse that is calculated by a covariance matrix (14) as follows.

$$\begin{pmatrix} x-\hat{x} & y-\hat{y} \end{pmatrix} \begin{pmatrix} \sigma_x^2 & \sigma_{xy} \\ \sigma_{xy} & \sigma_y^2 \end{pmatrix}^{-1} \begin{pmatrix} x-\hat{x} \\ y-\hat{y} \end{pmatrix} \leq D^2. \quad (28)$$

This equation indicates an error ellipse of Mahalanobis distance D, and the center of the ellipse corresponds to the estimated position ($\hat{x}$,$\hat{y}$) using an odometry system. The robot exists inside an error ellipse in a certain probability depending on Mahalanobis distance D. For example, in case that D is equal to 1, the probability that the robot exists in the error ellipse is 39.4%. The larger Mahalanobis distance extends, the larger the probability becomes. Please keep in your mind that we focus on a relation between a circular robot and obstacles, so the parameter of an orientational error does not have to be included to express this error ellipse.

4.4 Simulation of Positioning Error

We have simulated an estimation of positioning error using above method. We assume that each wheel radius ($\hat{R}_l$, $\hat{R}_r$) is 63[mm], and it's tread $\hat{T}$ is 399[mm]. Each standard deviation of the wheel (σ_{R_l}, σ_{R_r}) is 1.0[mm], and that of the tread σ_T is 1.0[mm]. Each initial value of the standard deviation of the positioning error is nothing. In that assumption, the robot traces a straight line 10[m], turns in 90[deg] and traces a straight line 10[m]. The result of this simulation is shown in Figure 2. The ellipses of "Mahalanobis distance=3" (an existence probability of a robot in each ellipse is equal to 98.9%) are drawn in this figure. The longer the robot navigates, the worse the estimated position becomes.

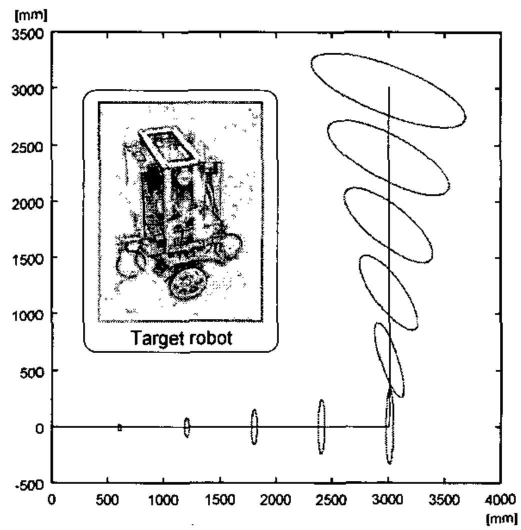

Figure 2: Simulation of positioning error by an odometry system

5 Path Evaluation

5.1 Evaluation of a Risk of Collision

To evaluate a risk of collision, we consider the estimated positioning error of a robot and positional relationship between the robot and obstacles. We define an evaluation method using error ellipse in equation (28), as follows.

First, we focus on the covariance matrix of the positioning error at a point x on the given path. Then, we increase Mahalanobis distance D of equation (28). It makes an error ellipse expand, and we continue this expansion until the error ellipse contacts an obstacle. In this paper, we call the ellipse that contacts to an obstacle at point x as "maximum ellipse", and we define "$D_{max}(x)$" as the parameter of Mahalanobis distance at point x. Figure 3 shows this idea. In this figure, Mahalanobis distance is increased until the error ellipse is contacted an obstacle. $D_{max}(x)$ is equal to the value of the Mahalanobis distance when the ellipse just contacts to the obstacle. Therefore, the robot is safe within a certain probability in the ellipse. The probability is theoretically calculated by Mahalanobis distance $D_{max}(x)$.

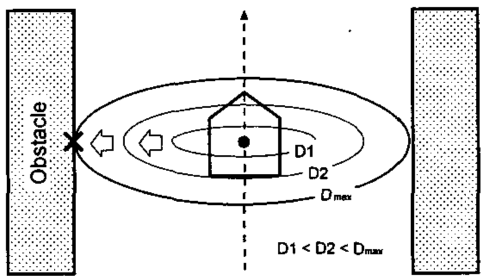

Figure 3: Evaluation of risk of collision

We consider that a risk of collision strongly relates to a "distance to obstacles" and "positioning uncertainty" at each position. Therefore we define a risk of collision $u(x)$ at each position x as following equation.

$$u(x) = \frac{1}{D_{max}(x)^2}. \tag{29}$$

In above equation, the bigger the probability of the existence, the smaller the risk of collision becomes. Conversely, if the probability of existence closes to 0%, the risk of collision $u(x)$ diverges to infinity.

On the other hand, the positioning error can be decreased by sensing landmarks using external sensor (E.g. by vision sensor). After adjustment of robot's position, the value of covariance matrix decreases, and an amount of the adjustment depends on sensing method. In our evaluation method, we assume that the variances of positioning error in equation (14) becomes a fixed value by sensing landmarks.

5.2 Evaluation of a Path

Next, we describe an evaluation of a risk of collision along a given path. We define an evaluation value of the path as a value of integral of a risk $u(x)$ along the robot's path. Therefore, evaluation value of the path U is expressed by a value of integral "$u(x)$", as follows.

$$U = \int_{Path} u(x)dx. \tag{30}$$

This means that a continuance of risky condition is not reasonable for a robot. The evaluation value becomes worse when a risky condition for the robot continues in its navigation.

5.3 Exclusion of Risky Path Candidates

In equation (30), the maximum value of $u(x)$ is not considered. However, it would be better not to select the path that includes extreme risky part temporarily, because the evaluation is considered as a value of integral. For example, let us think about a path that includes very narrow part. In this case, a risk of collision becomes high while the robot passes through the narrow part. However, evaluation value U in (30) doesn't always become large because U is a value of integral.

To avoid such absurdity, we add an idea to exclude paths that include extreme risky part. The maximum risk is calculated by $u(x)$ in (29), and its threshold is defined by a path planner. (If he thinks that the robot needs to take a safer path, the threshold should be smaller.)

5.4 Concrete Example of Our Evaluation Method

To clear the proposed evaluation method, we introduce a concrete example, shown in Figure 4. In this example, the robot simply goes straight along a corridor. In this figure, "standard ellipse" (Mahalanobis distance is constant) is drawn by broken line, and "maximum ellipse" is drawn by solid line.

When the robot does not use external sensors to adjust its position, monotonously a risk of collision $u(x)$ increases along the corridor (x1,x2,x3 in Figure 4). When the robot adjusts its position by detecting the landmark at the sensing point between x3 and x4, a risk of collision $u(x4)$ decreases to fixed value. That is because a variance of positioning error decreases, and it causes that $D_{max}(x)$ increases.

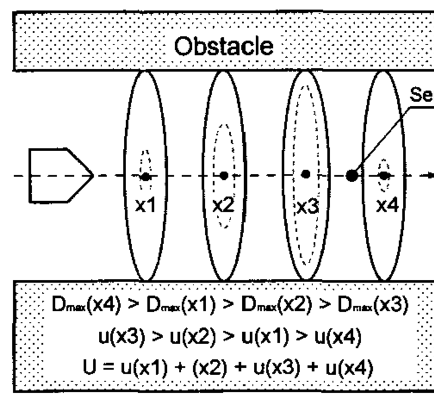

Figure 4: Evaluation of risk of collision

6 Example of Path Evaluation

We applied proposed evaluation method to two types of environment. One has no sensing point, and the other has several sensing points. In these examples, our objective is to compare paths' evaluation. Therefore, the idea of "Exclusion of Risky Path Candidates" (described in section 5.3) isn't considered.

6.1 Path Generation Using GVG

To apply proposed evaluation function to robot's path, we use GVG(Generalized Voronoi Graph)[2][5] to generate candidates of path. In case of two-dimensional environment, GVG is a set of points equidistant from two or more closest objects, expressed by following equation.

$$G(x) = \left[d_i - d_j\right](x) = 0 \tag{31}$$

where $d_i(x)$ is the closest point within an obstacle i from robot's position x. An example of GVG is shown in Figure 5.

Usually a start point and a goal point are connected to the closest GVG point on GVG edges. To simplify the problem, we set each point at a tip of GVG edges in this example. Sensing points are connected to the closest GVG point on GVG edges.

Each GVG edge is the path candidate that is the furthest from obstacles. Therefore, these are reasonable paths for our path evaluation method.

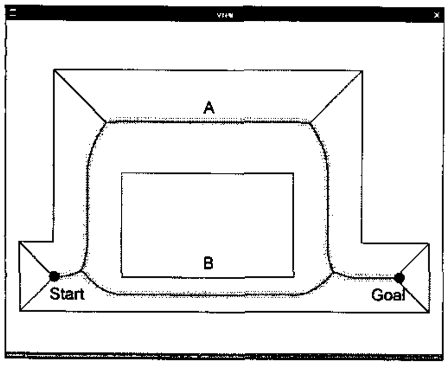

Figure 5: Path generation using GVG

6.2 Target Robot

In following examples, target robot's model is the same as the robot in section 4.4. Initial values of the error vector of the standard deviation ($\sigma_x, \sigma_y, \sigma_\theta$) are (1.0[mm], 1.0[mm], 0.1[rad]), and these are non-correlated.

6.3 Evaluation Example 1 : Path Evaluation Using only Odometry

First, we compare two evaluation values based on a risk of collision in two different paths from a start to a goal point in Figure 5. In this environment, "PATH A" is longer but wider than "PATH B".

Figure 6-(A1) shows "standard ellipse" (Mahalanobis distance is equal to 1) of "PATH A", and Figure 6-(B1) shows "standard ellipse" of "PATH B". Both are figures of reference that an estimated position of the robot becomes worse along both paths. Figure 6-(A2) and (B2) show "maximum ellipse" for a calculation of each path evaluation. Each ellipse is expanded until a part of it contacts obstacles. Please keep in your mind that the Mahalanobis distance $D_{max}(x)$ of each ellipse in (A2) and (B2) is various number, and a path evaluation is calculated by equation (29).

By applying our method, the evaluation value U of PATH A is 17.85, and that of PATH B is 33.73. From these results, "PATH A" is better than "PATH B" from the point of view of safety.

6.4 Position Estimation by Vision Sensor

Next, we consider a case of environment that includes sensing points for position adjustment of mobile robots. We assume that a robot can adjust its position by sensing landmarks of ceiling image. It is reasonable in indoor environment, and this method is already working well in our laboratory.

In case that the robot uses this method, its position and orientation can be adjusted by one sensing motion. However, this method also generates a positioning error caused by a sensing error. Therefore, we assume that the variance values of positioning error become fixed value by this motion.

6.5 Evaluation Example 2 : Path Evaluation with sensing

Next example is shown in Figure 7. It includes two sensing points that are located on PATH A in Figure 5.

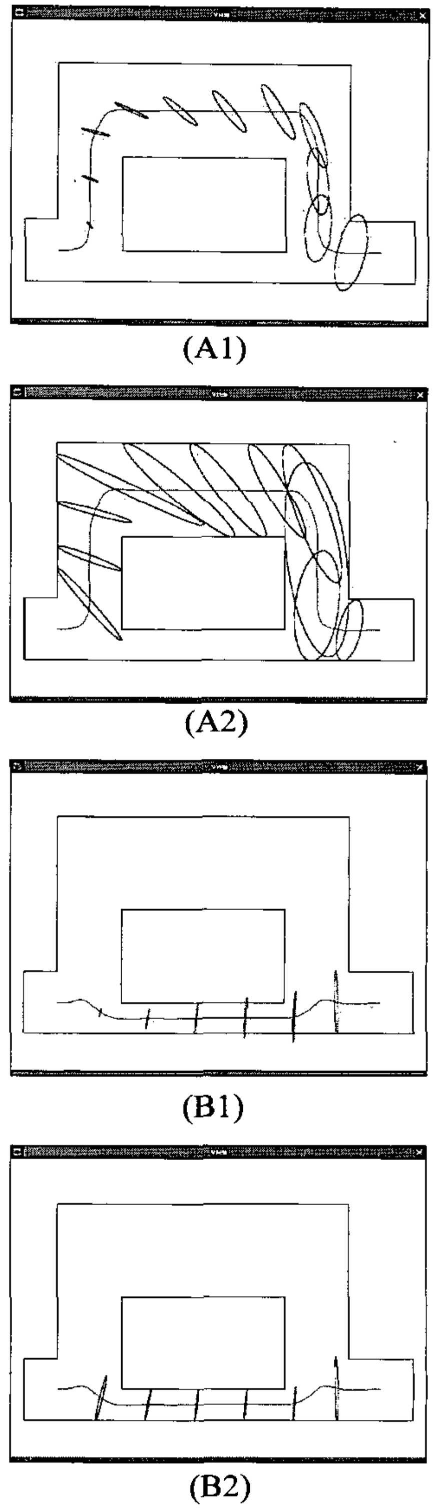

Figure 6: Example of path evaluation 1

These sensing points are named SP1 and SP2.

In these sensing points, we assume that the robot adjusts its position by detecting a ceiling image, described in section 6.4. Values of the error vector of the standard deviation $(\sigma_x, \sigma_y, \sigma_\theta)$ after sensing motion are (1.0[mm], 1.0[mm], 0.1[rad]), and these are non-correlated.

Table 1 shows the evaluation results of example 2. The evaluation result of this example shows that the path with sensing at SP2 only is better than at SP1 only or at both sensing points. The reason is that the robot accessing to SP1 is too close to the obstacle, then a risk of collision increases in case of sensing at SP1.

From above result, our evaluation method can exclude inappropriate paths (e.g. a path with risky sensing point) based on safety.

Table 1: The result of path evaluation 2

path	evaluation value
non-sensing	17.85
only SP1	7.41
only SP2	5.14
SP1 & SP2	6.32

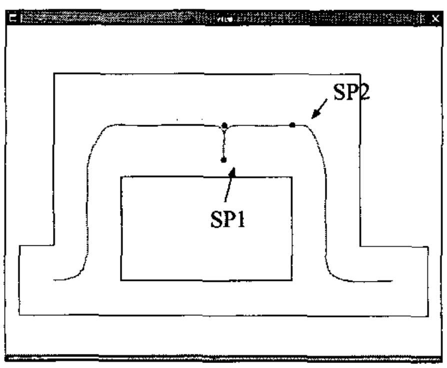

Figure 7: Example of path evaluation 2

7 Conclusion

In this paper, we proposed "path evaluation method based on a risk of collision" in known environment. Each path is evaluated by not only a relation between a robot and obstacles, but positioning uncertainty of the robot that is calculated from its internal parameters. Sensing point planning is also considered with the same frame of reducing a risk of collision. We also showed examples of path evaluation to verify a validity of the method.

In current evaluation method, uncertainty of robot's position is not considered when the robot arrives at goal point and sensing points. In future works, we will add an idea to solve this issue for path evaluation.

References

[1] J.Pan A.Kosaka. Purdue experiments in model-based vision for hallway navigation. In *Proc. of IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, pages 87–96, 1995.

[2] Howie Choset, Ilhan Konukseven, and Joel Burdick. Mobile robot navigation: Issues in implementation the generalized voronoi graph in the plane. In *Proc. of IEEE/MFI*, pages 241–248, Washington DC, 1996.

[3] I.H.Moon, J.Miura, Y.Yanagi, and Y.Shirai. Planning of vision-based navigation for a mobile robot under uncertainty. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems(IROS)'97*, pages 1202–1207, 1997.

[4] Kiyoshi Komoriya, Eimei Oyama, and Kazuo Tani. Planning of landmark measurement for the navigation of a mobile robot. In *Proc. of IEEE/RSJ International Conf. on Intelligent Robots and Systems*, pages 1476–1481, 1992.

[5] Jean-Claude Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1996.

[6] Syoichi Maeyama, Akihisa Ohya, and Shin'ichi Yuta. Non-stop outdoor navigation of a mobile robot – retroactive positioning data fusion with a time consuming sensor system –. In *Proc. of IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, pages 130–135, 1995.

[7] Keiji Nagatani and Shin'ichi Yuta. Path and sensing point planning for mobile robot navigation to minimize the risk of collision. In *Proc. of IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, pages 2198–2203, 1993.

[8] N.Roy, W.Burgard, D.Fox, and S.Thrun. Coastal navigation - mobile robot navigation with uncertainty in dynamic environment. In *Proc. of IEEE International Conf. on Robotics and Automation*, pages 35–40, 1999.