

C-space Exploration Using Noisy Sensor Models

Michael Suppa*, Pengpeng Wang[†], Kamal Gupta[†], Gerd Hirzinger*

*Institute of Robotics and Mechatronics

German Aerospace Center (DLR)

82230 Wessling, GERMANY

{Michael.Suppa, Gerd.Hirzinger}@dlr.de

[†]School of Engineering Science

Simon Fraser University

Burnaby, BC, CANADA, V5A 1S6

{pwangf, kamal}@cs.sfu.ca

Abstract—The concept of C-space entropy as a measure of knowledge of C-space for sensor-based path planning and exploration for general robot-sensor systems was recently introduced in [4]. The robot plans the next sensing action to maximally reduce the expected C-space entropy, also called the maximal expected entropy reduction, or MER criterion. The expected C-space entropy computation, however, made an idealized assumption. The sensor was assumed to measure exact data, i.e., it was not subject to noise. In this paper we extend this approach by using a real noisy sensor model. Sensing actions can then be compared on the basis of their uncertainty models. This offers the ability for using more than one principle sensor (multisensory exploration), because sensor readings can be weighted by evaluating the expected measurement quality. Additionally, it makes robot motion planning viable for tasks such as object surface inspection, which require the robot to come very close to the obstacles to achieve high sensing accuracy.

I. INTRODUCTION

Recent work at the Robotics Lab, Simon Fraser University, Canada has focussed on sensor-based motion planning for robots with non-trivial geometry and kinematics.¹ The problem here is to plan the next best view (NBV, or simply view planning), in order to efficiently explore the robot’s initially unknown environment. In order for the robot to be able to manoeuvre itself, the sensing action must be viewed from a configuration space (C-space) point of view, i.e., it must efficiently explore (make known) the robot’s C-space. The notion of C-space entropy was developed as a measure of the ignorance of C-space. Then the NBV is chosen so as to maximally reduce the expected C-space entropy. This is called maximal expected entropy reduction, or MER, criterion [4]. Algorithms based on MER criterion were developed [4], [8] and tested for both 2-D simulations and real experiments with the SFU eye-in-hand system, a PUMA 560 with wrist mounted range sensor. The results [10] showed great improvement over previous physical space based view planning algorithms.

⁰This work was done while the first author was doing visiting PhD research at School of Engineering Science, Simon Fraser University, Canada.

¹For sensor-based motion planning problems, the requirement — that the physical region has to be sensed free before the robot can actually occupy it — raises some fundamental novel issues that do not arise in the model-based case. See [7] for details.

However, in the above work, the sensor model used was assumed to measure exact data while real sensors are subject to noise. Therefore, more caution must be taken using results from this ideal sensing process, e.g. a safety margin to obstacles must be added.

At the Institute of Robotics and Mechatronics, Germany, a hand-guided multi-sensory device has been developed [1]. It consists of four different types of sensors: a long range but less accurate laser-stripe range sensor (lss), a short range but highly accurate laser-range scanner (lrs), a passive stereo vision sensor, and a texture sensor. The multi-sensory device is shown by itself in Fig. 1; and mounted on a passive 7-dof robotic manipulator (a passive Eye-in-Hand system), in Fig. 2. Experiences gained in the development of this passive Eye-in-Hand System led to the idea of automating the execution of inspection tasks, such as complete surface acquisition in 3-D environments, via an active Eye-in-hand system with the same multisensory device, but the passive 7-dof arm replaced by a 7-dof robot manipulator. For such capabilities, on the one hand, efficient algorithms are needed for the robot sensor system to explore its manoeuvrable space, the C-space, especially that corresponding to sensors’ positions around the object to be inspected; on the other hand, the robot is required to safely go very close to object for its “eye” to achieve high sensing accuracy, a must for surface inspection tasks.

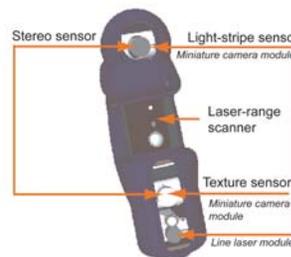


Fig. 1. Sensor principles integrated in the DLR hand-guided device

Fig. 2. Experimental setup: Hand-guided device mounted on a 7-Dof passive manipulator

These requirements of the automated inspection task mo-

tivate the idea of modelling the sensor noise in the planning stage to achieve much more accurate and reliable planning results. This will enable the robot to safely go much closer to the obstacles. Additionally, if we are using multiple sensors for the exploration task (multisensory exploration), a noisy statistical sensor model enables not only merging multiple readings (sensor fusion) but also decisions such as where to scan and which sensor to choose for view planning purpose, thus providing one integrated framework.

In this paper, we incorporate noisy sensor models in the C-space entropy computations for view planning. This serves as a first step for integrating C-space based view planning with surface inspection tasks. The computations of the noisy sensor probability distributions are inspired by the theory of Occupancy Grids (OC-Grids). This approach was developed for sonar sensors (which have a high angular sensing uncertainty) for creating maps of the environment of mobile robots [2]. The resulting probability distributions are then incorporated in the C-space entropy computations for view planning.

The paper is organized as follows. We first present a short background on C-space entropy and MER criterion, and expected entropy reduction computation using ideal noiseless sensor models. Then the noisy sensor model is presented, followed by modified expected entropy reduction computation. While emphasizing that the results are applicable to many degree of freedom (dof) robotic exploration tasks, for presentation and visualization purpose, we present test results of 2-D simulations for NBV problems. We use 2-D simulations mainly for ease of visualization and preliminary verification before actual implementation on a real robot-sensor system, currently under progress. Finally, the work is set in the context of a robotic inspection task.

II. BACKGROUND: C-SPACE ENTROPY

For sensor based motion planning, we may view the obstacle distribution in physical space as being derived from a spatial stochastic process (for example, a Poisson Point Process). This obstacle distribution in the physical space induces a probability distribution of possible C-space instances (each instance corresponds to one set of the statuses of all the unknown robot configurations), thereby the obstacle distribution in C-space would also be governed by a stochastic process. A measure of the knowledge (strictly speaking, lack of knowledge or ignorance), of this process is provided by Shannon's entropy, and is called the C-space entropy. The NBV (or next best sensing action) is then chosen to gain maximal expected knowledge of C-space, i.e., to maximally reduce C-space entropy, also called MER criterion. Here we briefly recapitulate the MER criterion after some notations. The robot is denoted by \mathcal{A} , the sensor by \mathcal{S} . The physical space is stated \mathcal{P} , whereas the unknown part of the physical space is denoted by \mathcal{P}_u . A robot configuration, or a point in the robot's configuration space \mathcal{C} , is denoted by q . The kinematics and geometry of the robot are captured in function $\mathcal{A}(q)$, which maps a point q in C-space to the region in \mathcal{P} occupied by the robot at q . The C-space entropy, $H(\mathcal{C})$,

is given by:

$$H(\mathcal{C}) = - \sum_{Q_1=0,1} \dots \sum_{Q_n=0,1} Pr[Q_1, \dots, Q_n] \log Pr[Q_1, \dots, Q_n] \quad (1)$$

Q_i denotes the random variable corresponding to a configuration q_i being free (=0) or in collision (=1), while n is the number of robot configurations in C-space, discretized with an appropriate resolution. $Pr[\cdot]$ denotes the probability of the corresponding outcome. One can now compute the expected information gain (or entropy reduction) $\Delta H(\mathcal{C})$ after sensing a region $\mathcal{V}(s) \in \mathcal{P}$ (obstacle/free). Here, we use s to denote a sensor's configuration that completely determines a sensing action²; and $\mathcal{V}(s)$ is the region to sense at s . The expected information gain (IG) is formulated as

$$IG_{\mathcal{C}}(s) = -E\{\Delta H(\mathcal{C})\}.$$

where E denotes the expectation operation. In [8], explicit closed form expressions were derived for $IG_{\mathcal{C}}(s)$ for an ideal beam sensor, assuming a Poisson point process model [11] for obstacles in the physical space³ and ignoring mutual information terms (for simplicity of computations). A beam sensor is characterized by a sensing ray of length L starting from the sensor origin. it gives the distance of the closest obstacle point (called hit point) along the beam. Fig. 3 shows

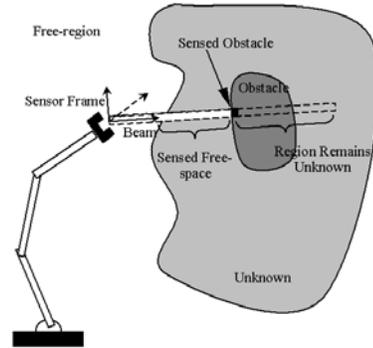


Fig. 3. The ideal beam sensor model without sensory noise.

the beam sensor model. Note that it has a zero volume field of view (FoV), and hence we compute the information gain density (IGD) [4], [6], [8]. The final expression of \widehat{IGD} , the approximation of IGD that omits mutual information, for the ideal beam sensor is given by

$$\widehat{IGD}_{\mathcal{C}} = \sum_{q \in \chi_u(s)} igd_q(s) = \frac{-\lambda}{L} \cdot \sum_{q \in \chi_u(s)} len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot \log(1-p(q)) \quad (2)$$

where $igd_q(s)$ describes the information gain density due to a configuration q ; $\mathcal{V}(s)$ denotes the sensing beam; and $\mathcal{V}_u(s)$ is the unknown part of $\mathcal{V}(s)$ that is in front of the first known obstacle along the sensing direction. $\chi_u(s)$, the unknown "C-zone" of the sensing beam s , is defined as the set of configurations whose collision status is unknown and at

² s would be, in general, different than q

³The Poisson Point model is characterized by uniformly distributed points (obstacles) in the physical space. Under this model, the probability of an arbitrary set $\mathcal{B} \in \mathcal{P}$ being free of obstacles can be described by $p(\mathcal{B}) = Pr[\mathcal{B} \subseteq \mathcal{P}_{free}] = e^{-\lambda \cdot vol(\mathcal{B})}$, where λ is a parameter representing the spatial density of obstacles.

which the robot has intersection with $\mathcal{V}_u(s)$. $p(q)$ denotes the probability that configuration q is collision free, and is given by

$$p(q) = e^{-\lambda \cdot vol(\mathcal{A}_u(q))}, \quad (3)$$

where $\mathcal{A}_u(q)$ is the unknown part of $\mathcal{A}(q)$.

III. C-SPACE ENTROPY COMPUTATION USING SENSOR UNCERTAINTY

A. Sensor Uncertainty Model

We now present the sensor uncertainty model to be used in the context of C-space exploration. This model was developed for the range sensors in the hand-guided multisensory data acquisition device (mentioned in the introduction) at DLR [1]. The laser-range scanner (lrs) has a very limited sensing range but high accuracy, while laser-stripe sensor (lss) has longer range but lower accuracy. Without loss of generality, we will limit our approach to these two different sensor uncertainty models. A general sensor uncertainty model (omitting specular reflections, because we deal with sensors based on triangulation not on time-of-flight principle) characterizing both models can be denoted as follows. We take the system view of this sensor uncertainty model. That is, the input of this model is processed by the model and the output is corrupted by noises. Both input and output are related by the following conditional probability density function (pdf). The conditional pdf of a noisy output saying that the obstacle is at position $r = x$,⁴ given the input saying that the obstacle is at position x_{obs} , ($x_{obs} \in [r_{min}, r_{max}]$ where r_{max} denotes the maximum sensing range and r_{min} denotes the minimum range.), denoted by $p(r = x|x_{obs})$, is given by a Gaussian distribution whose mean is x_{obs} ,

$$p(r = x|x_{obs}) = \frac{1}{\sqrt{2\pi}\sigma_r(x_{obs})} e^{-\frac{1}{2}\left(\frac{(x-x_{obs})^2}{\sigma_r^2(x_{obs})}\right)} = N(x_{obs}, \sigma_r(x_{obs})) \quad (4)$$

where the variable σ_r characterizes the radial variance of the sensor and $N(x_{obs}, \sigma_r(x_{obs}))$ denotes a Gaussian distribution with mean x_{obs} and variance $\sigma_r(x_{obs})$. (For now, because of complexity, no angular variance is taken into account for the laser-range scanner model, even though the sensor has an angular uncertainty of 0.9° .) Similarly, we have the conditional pdf of a noisy output saying that the obstacle is at position $r = x$, given that the input saying the beam is free of obstacles, denoted by $p(r = x|\neg x_{obs})$, as follows,

$$p(r = x|\neg x_{obs}) = \frac{1}{\sqrt{2\pi}\sigma_r(r_{max})} e^{-\frac{1}{2}\left(\frac{(x-r_{max})^2}{\sigma_r^2(r_{max})}\right)} = N(r_{max}, \sigma_r(r_{max})) \quad (5)$$

Both lrs and lss sensors are based on the triangulation principle, therefore, the uncertainty of the measurements (variance σ in Eq. 4) is dependent on the obstacle's real distance x_{obs} . A simple yet powerful model [1] to characterize this dependence is given by,

$$\sigma_r(x_{obs}) = k \cdot (x_{obs})^i, \quad (6)$$

⁴All the distances are computed with respect to the sensor's origin.

As implied by this equation, the sensor will give more accurate result (lower variance) when the obstacle is nearer. The statistical analysis of the laser-range scanner [1] results in the following numerical parameters for k and i :

$$k_{lrs} = 5.52 \cdot 10^{-9} / \text{mm}^3 \text{ and } i_{lrs} = 4.01 \approx 4$$

The laser-range scanner uncertain model is geometrically illustrated in Fig. 4. It is derived in the local sensor coordinate system CoS_{sensor} . The variable ω denotes the angular velocity of the laser-range scanner (See [1] for details), while α_{on} and α_{off} denote the angles at which the laser is turned on and turned off respectively, limiting the FoV.

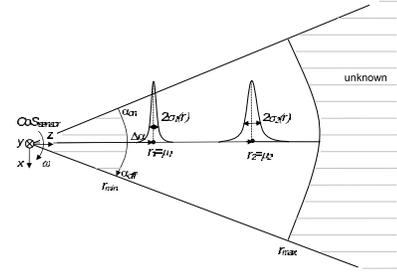


Fig. 4. Probability distribution given by a laser-range sensor uncertainty model.

Using the same techniques, we gain a similar model as in Eq. 4 for laser-stripe sensor, however it has a different accuracy (variance over sensing distance) as a function of the radial measurement. Admittedly, a more appropriate sensor uncertainty model will be more complex.⁵

B. C-space Entropy Computation Based on Beam Sensor Model

Now we outline the derivation of the C-space entropy computation applying the sensor uncertainty model described above to a beam sensor. Note that there is key difference here from the map building literature in mobile robots (For further reading on this topic, we refer the reader to [3]). There, after new sensory data is acquired, one could use Bayes Rule or Dempster-Shafer theory for integrating these new measurements, based on sensor uncertainty models, into existing maps. These sensor uncertainty models deliver an estimation for the probability of the state (obstacle/free) of a cell in the environment, after obtaining an actual sensor reading. In view planning, no actual sensor reading is made yet; one is trying to plan where to view next, which entails determining the expected outcome of a “potential” sensor reading based on certain probabilistic assumptions about the environment (e.g., Poisson Point process model for obstacle distribution), a kind of “inverse computation” to that of building a world map. To clarify this approach, we first deal with the model for an ideal beam sensor without noise, which can tell the obstacle's

⁵The laser stripe sensor consists of a camera system, thus even though we make use of rectified camera images, the accuracy is dependent not only on the distance, but also on the angle relative to the center of the camera. Recent work at DLR deals with the derivation of a more accurate model for the laser-stripe sensor.

real position within the sensing beam. (Again, we take the system point of view.) There are two cases. First, suppose we have an obstacle at position x_{obs} (the input). The output of this ideal sensor is a sensing result that the region in front of x_{obs} (seen from the sensor origin) is free, the position at x_{obs} is an obstacle, and the area behind x_{obs} remains unknown. Consider the world model of the beam as a continuous version of OC-Grids, i.e., each position x on the beam can either be free or obstacle. After sensing, the ideal sensor outputs a probability function over this world model, i.e., the probability of each position x being obstacle, under the condition that the obstacle is at x_{obs} , is given by, (We use δ in the following equation to denote an ideal sensor model.)

$$\Pr_{world}^{\delta}(r = x|x_{obs})(x) = \delta(x - x_{obs}) + 0.5u(x - x_{obs}) \quad (7)$$

In the above equation (x) denotes the probability is a function of position of x and $r = x$ denotes this is the world model at position x . $\delta(x - x_{obs})$ denotes an impulse function at position x_{obs} and $u(x - x_{obs})$ denotes a step function at x_{obs} . Otherwise, if there is no obstacle in the sensor range, the sensory output tells that the beam is free. The probability function over the world model under the condition that the beam is free of obstacles is given by,

$$\Pr_{world}^{\delta}(r = x|\neg x_{obs})(x) = 0.5u(x - r_{max}) \quad (8)$$

Again, the sensing range is $r_{min} < x < r_{max}$.

The noisy sensory data can be thought of as a blurring process, where an ideal sensing is processed by a noisy system, given in Sec. III-A. By basic signal and system theory [13], the result is a convolution of the ideal sensor result with the Gaussian distribution given in Eq. 4, whose mean is the position of the actual obstacle. That is, given the condition that an obstacle is at position x_{obs} , the probability of the world model is given by, (In the following equation, we use G to denote it is a sensor with Gaussian noise.)

$$\begin{aligned} \Pr_{world}^G(r = x|x_{obs})(x) \\ = \Pr_{world}^{\delta}(r = x|x_{obs})(x) * N(0, \sigma_r(x_{obs})) \\ = a(u(x - x_{obs}) + \delta(x - x_{obs})) * N(0, \sigma_r(x_{obs})) \end{aligned}$$

In the above equation, a denotes the a priori probability for an area being unknown; we assume $a = 0.5$ [2], using so called non-informative priors.

If the whole beam is free, the conditional probability of the world model is given by,

$$\begin{aligned} \Pr_{world}^G(r = x|\neg x_{obs})(x) \\ = \Pr_{world}^{\delta}(r = x|\neg x_{obs}) * N(0, \sigma_r(r_{max})) \\ = a \cdot u(x - r_{max}) * N(0, \sigma_r(r_{max})) \end{aligned}$$

Using the *error-function* $\mathbf{erf}()$, which describes the integral of a normal distribution, we have,

$$\begin{aligned} \Pr_{world}^G(r = x|x_{obs})(x) = a \left[\frac{1}{\sqrt{2\pi}\sigma_r(x_{obs})} e^{-\frac{1}{2} \frac{(x-x_{obs})^2}{\sigma_r^2(x_{obs})}} \right. \\ \left. + \frac{1}{2} (1 + \mathbf{erf}(\frac{\sqrt{2} \cdot (x - x_{obs})}{2 \cdot \sigma_r(x_{obs})})) \right] \end{aligned} \quad (9)$$

and

$$\Pr_{world}^G(r = x|\neg x_{obs})(x) = \frac{a}{2} (1 + \mathbf{erf}(\frac{\sqrt{2} \cdot (x - x_{obs})}{2 \cdot \sigma_r(x_{obs})})) \quad (10)$$

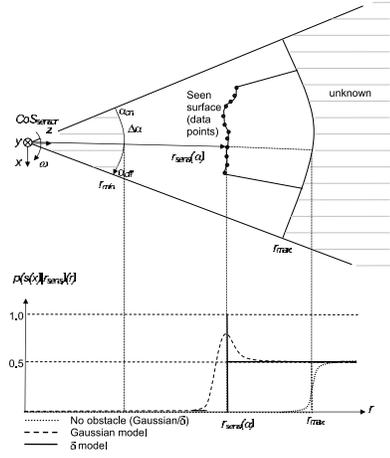


Fig. 5. Probability distribution assuming uncertain sensing

As stated at the beginning of this section, we still need a probability distribution assumption of the obstacles in the robotic environment and again we use Poisson point process. For view planning, we are interested in the “inverse computation”, the conditional probability of having an obstacle at position x_{obs} , given the probability function of the world model given by Eq. 9. By Bayes rule, we have,

$$\Pr(x_{obs}|world) = \int_{r_{min}}^{r_{max}} \Pr(x_{obs}) \Pr_{world}^G(r = x|x_{obs})(x) dx \quad (11)$$

In the above equation, we use $\Pr(x_{obs})$ to denote the obstacle probability given by Poisson point process. Similarly, we have the conditional probability that the beam is free, given the probability of the world model in Eq. 10 as follows,

$$\Pr(\neg x_{obs}|world) = \int_{r_{min}}^{r_{max}} \Pr(\neg x_{obs}) \Pr_{world}^G(r = x|\neg x_{obs})(x) dx \quad (12)$$

Now we can compute the information gain using formulae developed above. The information gain computation for the blurred beam sensor can be divided into three events. First, there is a hitpoint inside $\mathcal{A}_u(q) \cap \mathcal{V}(s)$, the unknown part of the robot at configuration q inside the unknown part of FoV with sensing action s . Secondly, we have a hitpoint outside $\mathcal{A}_u(q) \cap \mathcal{V}(s)$ and finally we do not have a hitpoint within the measurement range. The following equations formulate this division of computing expected values of entropy reduction at different cases and corresponding information gain densities. (Details of this method for ideal noiseless sensor can be found in [8])

$$\begin{aligned} E_G\{\Delta H(Q)\}_s = E\{\Delta H(Q)\}_{x_{obs} \in \mathcal{A}_u(q)} + E\{\Delta H(Q)\}_{x_{obs} \notin \mathcal{A}_u(q)} + E\{\Delta H(Q)\}_{\exists no\ obs} \quad (13) \\ \Leftrightarrow (igd_q) = (igd_q)_1 + (igd_q)_2 + (igd_q)_3 \end{aligned}$$

We present final expressions for each of the terms in Eq. 13 using Eq. 11 and Eq. 12. The derivation is omitted for lack of space.

The first event corresponds to there is a hitpoint (the real position of the obstacle) inside $\mathcal{A}_u(q) \cap \mathcal{V}(s)$. So we know for sure q is in collision after sensing. The entropy reduction for this q will be its entropy $H(Q)$, given by

$$H(Q) = -p(q) \log(p(q)) - (1 - p(q)) \log(1 - p(q)) \quad (14)$$

The information gain density is given by:

$$(igd_q)_1 = \frac{\lambda H(Q)}{M_{obs}} \int_{x \in \mathcal{A}(q) \cap \mathcal{V}_u(s)} \int_{r_{min}}^{r_{max}} \Pr^G_{world}(r = x|x_{obs})(x) dx_{obs} dx \quad (15)$$

and for normalization use, we have,

$$M_{obs} = \int_0^{r_{max} + 2 \cdot \sigma_{max}} \int_{r_{min}}^{r_{max}} \Pr^G_{world}(r = x|x_{obs})(x) dx_{obs} dx \quad (16)$$

The second event is when the hitpoint is outside $\mathcal{A}_u(q) \cap \mathcal{V}(s)$, but inside $\mathcal{V}_u(s)$. The entropy reduction is induced by the addition free region sensed (the part of the $\mathcal{V}_u(s)$ in front of hitpoint). The result is given by:

$$(igd_q)_2 = 0 \quad (17)$$

Note that in this event, even if the entropy reduction density (entropy reduction divided by the beam volume and compute the limit when this volume approaches to zero) is not zero. Poisson point assumption tells us the expectation (even after being blurred by the sensory noise) is zero.

The last event is when the whole sensing beam is sensed free. The result is given by:

$$(igd_q)_3 = \frac{1}{M_{free}} \frac{dH(Q)}{dV} \int_{x \in \mathcal{A}(q) \cap \mathcal{V}_u(s)} \int_{x_{min}}^{x_{max}} \Pr^G_{world}(r = x|\neg x_{obs})(x) dx_{obs} dx \quad (18)$$

and for normalization use, we have

$$M_{free} = \int_0^{r_{max} + 2 \cdot \sigma_{max}} \int_{x_{min}}^{x_{max}} \Pr^G_{world}(r = x|\neg x_{obs})(x) dx_{obs} dx \quad (19)$$

The term $\frac{dH(Q)}{dV}$, using chain rule, is given by,

$$\frac{dH}{dV} = \frac{dH}{dp} \frac{dp}{dV} = \underbrace{(-\log(p(q)) + \log(1 - p(q)))}_{\frac{dH}{dp}} \cdot \underbrace{(-\lambda p(q))}_{\frac{dp}{dV}} \quad (20)$$

By using all the above equations in the IGD computation in Eq. 2, we can get the result of the information gain for each sensing action s .

By comparing with the result given by ideal sensor model, we can see the additional integral term (the inner integration) over the range $[r_{min}, r_{max}]$ in Eq. 15 and Eq. 18 induced by noise (The outer integration is the expectation computation of Poisson point process). This term depends on the accuracy of sensing, i.e., variance σ_r . As implied by Eq. 6, this accuracy is better for closer sensing range. This would imply that the value of information gain density will be smaller if sensing accuracy is lower, i.e., σ_r is higher. Therefore, the information gain density will be low for sensing range farther away as shown in Fig. 6. So in general, the view planning result will be a tradeoff between sensing accuracy and information gain density. As for view planning, the sensing action s will be chosen according to $s_{max} = \arg \max_s IGD(s)$. The pseudo code for the resulting algorithm is listed as follows,

FOR(all sensing action s) % according to a certain resolution

```
{
  IGD(s)=0 //Initialization
  FOR(all unknown Configurations q) {
    Compute unknown part of the robot at q, Au(q)
    Compute unknown part of FoV, Vu(s)
    IF(Au(q) intersects Vu(s)) {
      Compute H(Q) and dH(Q)/dV
      Numerically compute (igdq)1, (igdq)2 and (igdq)3
      Add information gain for q to IGD(s) i.e.
      igdq = (igdq)1 + (igdq)3
      IGD(s) += igdq
    }
  }
} Choose s_max=argmax IGD(s) as the view planning result
```

In this algorithm, the basic computations are simple geometric ones: calculation of $\mathcal{V}_u(s)$ for each s , calculation of $\mathcal{A}_u(q)$ for each unknown configuration q , and the intersection between them. Having computed these, the probability of configuration q being free, and the corresponding entropy, $H(q)$, are easily calculated from Eq. 3 and Eq. 14. $\Pr^G_{world}(r = x|x_{obs})(x)$ and $\Pr^G_{world}(r = x|\neg x_{obs})(x)$ in equations, Eq. 9 and Eq. 10, are computed numerically (because of lack of closed form expression) using a MacLaurin series expansion of the error function up to 6th order [14]. In the implementation, these are stored in look-up tables, to allow for fast computations. Having computed these basic entities, $(igd_q)_1$ and $(igd_q)_3$ in Eq. 15 and Eq. 18, are computed using numerical integration. We discretize both the inner and outer integrations with a step size $n = 5 \cdot \sigma_{min}$, where $\sigma_{min} = k \cdot (r_{min})^i$ is the minimum variance.

IV. EXPERIMENTS

We now present some preliminary 2-D simulation results with a planar two-link robot equipped with a simulated laser range scanner (lrs) and laser strip sensor (lss) and show how the two different sensors perform for C-space exploration.

Fig. 7 shows a typical simulation environment. The task of the robot is to start from the initial configuration (vertically down), and to explore the region around it to get maximum knowledge of the C-space. The white region surrounding the initial robot configuration is assumed free, the light grey region is unknown, and dark gray regions are obstacles unknown to the robot. The range sensor FOV is the triangle shown in the figure. Two types of sensors, one with shorter range but high accuracy and one with longer range but low accuracy, are simulated for lrs and lss sensors respectively. The specifications of both sensor are in Table I. Obstacles are detectable in the range $[r_{min}, r_{max}]$. The simulation environment as well as its data structures are similar to those described in detail in [12].

We use the tuple (α, β, θ) as the sensor's configuration s , where α denotes the first joint angle of the robot, β denotes the second joint angle, and θ denotes the angle between the medial axis of sensor FoV and the robot's second link. The view planning algorithm returns s_{max} that gives the maximum value of $IGD(s)$. The medial axis of the sensor FoV is placed along this s_{max} for the next scan. Fig. 8 shows snapshots of the physical space and C-space after 18 scans. Fig. 9 displays the exploration rate comparison between these two sensor models. It is interesting to see that the exploration rates are not too bad comparing with that of ideal, noiseless sensor model as

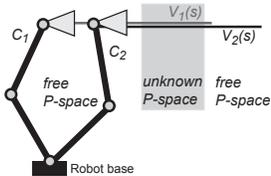


Fig. 6. Configurations C_1 and C_2 will produce the same IDG under ideal sensing. C_2 will be chosen since it leads more sensing accuracy.

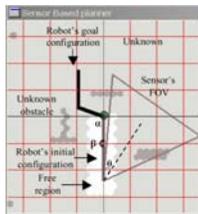
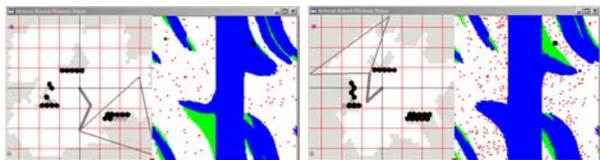


Fig. 7. An eye-in-hand system a two-link robot with a wrist mounted range sensor (with triangle FOV) moving in an unknown environment.

TABLE I
SENSOR PROPERTIES

Sensor:	r_{min}	r_{max}	α	$\sigma(r_{min})$	$\sigma(r_{max})$
Laser-range scanner	50 mm	300 mm	55	0.02 mm	4.73 mm
Laser-stripe sensor	80 mm	400 mm	30	0.05 mm	15.00 mm

in [8]. Intuitively, this is because the noise level in our current simulations is small compared with resolution needed for gross motion planning. For realistic inspection tasks, this would not be the case. Our current efforts are to develop such test cases and test our view planning algorithm on them.



(a) Laser-stripe sensor (b) Laser-range scanner

Fig. 8. Physical space and C-space after 18 iterations, P-space is idealized

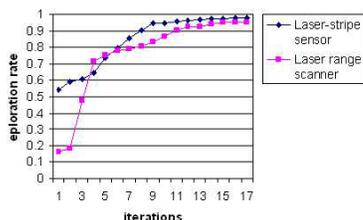


Fig. 9. Comparison of Exploration results: known C-space percentage vs. number of iterations

V. FRAMEWORK FOR AUTONOMOUS INSPECTION

Our ultimate goal is an autonomous robotic system that is capable of multisensory surface inspection tasks. We will implement the C-space exploration algorithm described in the preceding sections in order to gain high manoeuvrability in the context of an object inspection task. The general framework of such a system is illustrated in Fig. 11 and a putative representation or the data structure of the physical space in the context of both C-space exploration and object inspection is shown in Fig. 10. In this framework, a region of interest (RoI), e.g., a sphere region with center \underline{c}_{RoI} and radius r_{RoI} is first identified. This region of interest could either be provided by e.g. passive stereo vision based on edge detection, or, by a priori information about the physical workspace. Assuming

that \underline{c}_{RoI} and r_{RoI} thus roughly known, C-space exploration will then be carried out.

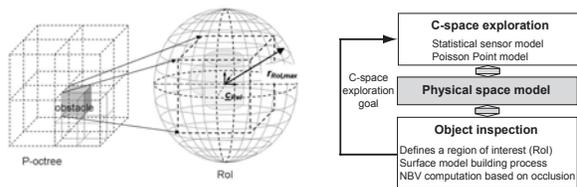


Fig. 10. Future implementation: world model for the C-space exploration on the left, the closeup for the inspection task on the right

VI. CONCLUSION AND FUTURE WORK

In this paper, we used a sensor noise model to compute C-space entropy in the context of C-space exploration. This work was motivated by a need for completely automated inspection tasks with multiple and noisy real range sensors. 2-D simulation shows promising results. We would like to integrate these preliminary results into an integrated framework for autonomous inspection and view planning. This will include a many dof simulation and experiments on a real robot.

REFERENCES

- [1] Suppa, M., Hirzinger, G. *A Novel System Approach to Multisensory Data Acquisition*, The 8th Conference on Intelligent Autonomous Systems IAS-8, Amsterdam, The Netherlands, 2004.
- [2] Elfes, A., *Occupancy Grids: A Probabilistic Framework for Mobile Robot Perception and Navigation*, PhD thesis, Electrical and Computer Engineering Dept./Robotics Inst, Carnegie Mellon University, 1989.
- [3] Dorst, L., van Lambalge, M., Voorbraak, F., *Reasoning with uncertainty in robotics*, RUR 95, Lecture Notes in Computer Science 1093, Springer, 1996.
- [4] Yu, Y., Gupta, K., *An Information Theoretic Approach to Viewpoint Planning for Motion Planning of Eye-in-Hand Systems*, 31st International Symposium on Industrial Robotics (ISR) 2000, Montreal, Canada, 2000.
- [5] Yu, Y., Gupta, K., *Sensor-Based Probabilistic Roadmaps: Experiments with an Eye-in-Hand System*, Journal of Advanced Robotics, 2000.
- [6] Yu, Y., Gupta, K., *An Information Theoretical Approach to View Planning with Kinematic and Geometric Constraints*, International Conference on Robotics and Automation ICRA, 2001.
- [7] Gupta, K., Yu, Y., *On Eye-sensor Based Path Planning for Robots with Non-trivial Geometry/Kinematics*, International Conference on Robotics and Automation ICRA, 2001.
- [8] Wang, P., Gupta, K., *Computing C-Space Entropy for View Planning Based on Beam Sensor Model*, Proceedings of the International Conference on Intelligent Robots and Systems IROS 2002, Switzerland, 2002.
- [9] Wang, P., Gupta, K., *View Planning via Maximal C-Space Entropy Reduction*, Fifth International Workshop on Algorithmic Foundations of Robotics WAFR 2002, Nice, France, 2002.
- [10] Wang, P., *View Planning Via Maximal C-space Entropy Reduction*, Master's Thesis, School of Engineering Sciences, Simon Fraser University, April 2003.
- [11] Stoyan, D., Kendall, W.S., *Stochastic Geometry and Its Applications*, J. Wiley, 1995.
- [12] Ahuactzin, J., Portilla, A., *A Basic Algorithm and Data Structures for Sensor-based Path Planning in Unknown Environments*, Proceedings of the International Conference on Intelligent Robots and Systems IROS 2000, Takamatsu, Japan, 2000.
- [13] Oppenheim, A., Schafer, R., *Discrete time signal processing*, Prentice Hall, 2nd edition, 1998.
- [14] Jeffrey, A. *Handbook of Mathematical Formulas and Integrals*, San Diego : Academic Press, 1995.