
View Planning via Maximal C-space Entropy Reduction

Pengpeng Wang (pwangf@cs.sfu.ca)¹ and Kamal Gupta (kamal@cs.sfu.ca)¹

School of Engineering Science, Simon Fraser University, Burnaby, BC, CANADA, V5A 1S6

Abstract. We introduced the concept of C-space entropy recently in [1–3] as a measure of knowledge of C-space for sensor-based path planning and exploration for general robot-sensor systems. 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 two idealized assumptions. The first was that the sensor field of view (FOV) is a point; and the second was that no *occlusion (or visibility) constraints* are taken into account, i.e., as if the obstacles are transparent. We extend the expected C-space entropy formulation where these two assumptions are relaxed, and consider a generic range sensor with non-zero volume FOV and occlusion constraints, thereby modelling a real range sensor. Planar simulations show that (i) MER criterion results in significantly more efficient exploration than the naive physical space based criterion (such as maximize the unknown physical space volume), (ii) the new formulation with non-zero volume FOV results in further improvement over the point FOV based MER formulation.

1 Introduction

While most research in sensor-based path planning and exploration has concerned itself with mobile robots, our recent work has concentrated on general robot-sensor systems, where the sensor is mounted on a robot with non-trivial geometry and kinematics [1–3,5]. See also [6–10]. This class of robots is broad and includes robots ranging from a simple polygonal robot to articulated arms, mobile-manipulator systems, and humanoid robots [11]. The sensor is assumed to be an “eye” type sensor that is capable of providing distances from a given vantage point (actual implementation may be a laser range scanner, passive stereo vision, etc.). Figure 1 shows a simple example of such a robot-sensor system — an eye-in-hand system — an articulated arm with a wrist mounted range sensor. The robot must simultaneously plan paths and sense its environment for obstacles. Unlike for a simple mobile robot (often modelled as a point, see [13] for next best view planning for mobile robots), where the robot can move (path planning) and what it should sense (view planning), has a much more complex relationship here [5]. “Where to move” is best posed and answered in configuration space, the natural space for path planning. In [1,2], we showed that the view planning problem is also appropriately posed in the configuration space of the robot — the next view should

be planned to give maximum knowledge or information (whether a configuration is free or in collision with an obstacle) of the C-space of the robot. What this implies is that the sensing action, which obviously senses physical space, must be (implicitly or explicitly) transformed to the configuration space. Treating the unknown environment stochastically, we introduced the notion of C-space entropy, measure of the robot’s knowledge of C-space. The next best view is then the one that maximizes the expected entropy reduction (MER criterion) or, equivalently expected information gain. In contrast, earlier approaches had simply used a naive criterion, such as maximize unknown physical space volume (MPV) in the sensor FOV, to choose the next best view [6].

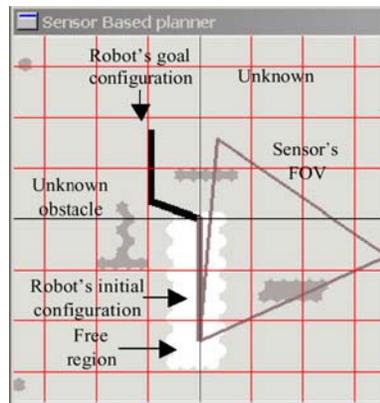


Fig. 1. An eye-in-hand system — a two-link robot with a wrist mounted range sensor (with triangle FOV) moving in an unknown environment. A key question (the view planning problem) is: where should the robot sense next?

We derived closed form expressions for expected C-space entropy reduction, or information gain under a Poisson point process model of the environment [12]. However, two idealized assumptions were made regarding the sensor in that paper: (i) the sensor has a point field of view (FOV), i.e., it senses a single point and (ii) no occlusion constraints were taken into account, i.e., as if the sensor would “see” (get range measurement) through the obstacles. The next best view is planned using this formulation, i.e., the algorithm computes the point (say, x_{max}) which, if sensed, would yield maximum expected information gain and places the sensor so that the center of the actual FOV (a cone) coincides with x_{max} .

In this paper, we relax the above two assumptions and present the MER criterion computation for a generic range sensor with a non-zero volume FOV while respecting occlusion constraints, thereby modelling real range sensors. This computation is valid for a Poisson point process model of the environment, admittedly a simplification, but the resulting closed form expressions give us insights and are useful at least as approximations. We present simulations that show clear improvement in the efficiency of exploration with the

new formulation. Our initial simulations are planar for ease of visualization. We emphasize that our formulations and results are valid for 3D environments and are currently being implemented on a real six-dof SFU eye-in-hand system consisting of a PUMA 560 with a wrist mounted area-scan laser range sensor that has been developed in our lab and was reported in [5]. We expect to report these experimental results in the near future.

2 Notation

Let \mathcal{A} denote the robot and q denote a point in its configuration space, \mathcal{C} . $\mathcal{A}(q)$ then denotes the region in physical space, \mathcal{P} , occupied by the robot. Let \mathcal{S} denote a sensor attached to the robot. We attach a coordinate frame to the sensor's origin. Let s denote the vector of parameters that completely determine the sensor frame, i.e., sensor's configuration. For instance, assuming the sensor is attached to the end-effector of the robot, for planar case, $s = (x, y, \theta)$; for 3D case, $s = (x, y, z, \alpha, \beta, \gamma)$. Let $\mathcal{V}(s) \in \mathcal{P}$ denote the region to be sensed (sensor FOV) by the sensor at configuration s . We discuss three different sensor FOV's: (i) an idealized point FOV sensor, i.e., $\mathcal{V}(s) = x$, a point $\in \mathcal{P}$, (ii) a beam FOV sensor, i.e., $\mathcal{V}(s) = L$, a line segment $\in \mathcal{P}$, and (iii) a non-zero volume FOV sensor, such as an area scan range sensor, i.e., $\mathcal{V}(s) = \mathcal{B}$, an open set $\in \mathcal{P}$. Subscripts *free*, *obs*,¹ and *u* denote the known free, known obstacle and unknown regions, respectively in physical and configuration space. So, for example, \mathcal{P}_{obs} denotes the known obstacles in physical space, $\mathcal{A}_u(q)$ denotes the part of robot lying in unknown physical space at configuration q , and \mathcal{C}_{free} denotes the known free configuration space.

3 C-space Entropy and MER criterion

In the following, first we overview the notion of C-space entropy and the closed form expression for point FOV sensor derived in [1,2], followed by the closed form expression for the beam FOV sensor. We then present the closed form expression for the generic FOV sensor.

We assume that the obstacles' distribution in the physical environment is modelled with an underlying stochastic process (e.g., the Poisson model used later). The kinematics and geometry of the robot, embodied by function $\mathcal{A}(q)$ map the probability distribution in physical space to a probability distribution over the C-space. Shannon's Entropy then provides a measure of the robot's ignorance of the status of C-space [2]. One can then compute the expected entropy reduction (or, equivalently, expected information gain) if a region $\mathcal{V}(s) \in \mathcal{P}$ was sensed (obstacle/free). The next best view (the region

¹ Ideally, these two subscripts should be *known-free* and *known-obs*. But we omit *known* for brevity.

to be sensed) is then the one that maximizes the expected entropy reduction (MER criterion), or equivalently, expected information gain (IG).

The information gain (IG) function is defined as

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

where $H(\mathcal{C})$ denotes the current C-space entropy, $E_s\{\Delta H(\mathcal{C})\} = E\{H(\mathcal{C} | \mathcal{V}(s))\} - H(\mathcal{C})$ denotes the expected entropy change after $\mathcal{V}(s)$ is sensed.

3.1 Environment Model

We use a simple probabilistic model of physical space — the Poisson point process, essentially characterized by uniformly distributed points in space [12]. From a motion planning point of view, these points, denoted by pt , are obstacles in the physical space of the robot. Given the density parameter of this model, λ , the void probability of an arbitrary set $\mathcal{B} \in \mathcal{P}$ — the probability that there is no pt (obstacle) in \mathcal{B} — denoted by $p(\mathcal{B})$, is given by

$$p(\mathcal{B}) = \Pr[\mathcal{B} \subseteq \mathcal{P}_{free}] = e^{-\lambda \cdot vol(\mathcal{B})} \quad (1)$$

3.2 Point FOV Sensor

For a point FOV sensor model, which only senses a point (or an infinitesimal ball) in physical space, rather than compute information gain, it is more appropriate to compute the corresponding density function, i.e., expected entropy reduction (or, equivalently, expected information gain) per unit volume if a point $x \in \mathcal{P}$ was sensed (obstacle/free). The reason is as follows [1,2]. The information gain occurs either because the status of an unknown point becomes obstacle or free. While the absolute information gain due to a single point becoming free is zero almost everywhere, the corresponding density function is non zero and finite! Thus, it gives a more complete picture of the information gain.

The information gain density (IGD) function for a point FOV sensor is defined as

$$IGD_{\mathcal{C}}(x) = \lim_{vol(\mathcal{B}(x)) \rightarrow 0} \frac{-E_{\mathcal{B}(x)}\{\Delta H(\mathcal{C})\}}{vol(\mathcal{B}(x))}$$

where, consistent with earlier notation, $H(\mathcal{C})$ denotes the current C-space entropy, $E_{\mathcal{B}(x)}\{\Delta H(\mathcal{C})\} = E\{H(\mathcal{C}|\mathcal{B}(x))\} - H(\mathcal{C})$ denotes the expected entropy change after $\mathcal{B}(x)$, a ball centered at point x , is sensed.

In order to get efficiency in computing, we neglect the mutual entropy terms², essentially treating each (discretized) configuration as an independent

² Please see section 6 for more discussion on this.

random variable, i.e., $\tilde{H}(\mathcal{C}) = \sum_{q \in \mathcal{C}} H(Q)$. In this equation, Q denotes the binary random variable (r.v.) corresponding to configuration q being free (=0) or in collision (=1); $H(Q)$ denotes the entropy of r.v. Q , i.e.,

$$H(Q) = p(q) \cdot \log(p(q)) + (1 - p(q)) \cdot \log(1 - p(q)) \quad (2)$$

where $p(q) = \Pr[q = \text{free}] = e^{-\lambda \cdot \text{vol}(\mathcal{A}_u(q))}$ is the marginal probability that configuration q is collision-free, also called the void probability of q . With this simplification one can show that:

$$\widetilde{IGD}_{\mathcal{C}}(x) = \lim_{\text{vol}(\mathcal{B}(x)) \rightarrow 0} \frac{-E\{\Delta \tilde{H}(\mathcal{C})\}_{\mathcal{B}(x)}}{\text{vol}(\mathcal{B}(x))} = \sum_{q \in \mathcal{C}} \text{igd}_q(x)$$

where $\text{igd}_q(x)$ is given by:

$$\text{igd}_q(x) = \lim_{\text{vol}(\mathcal{B}(x)) \rightarrow 0} \frac{-E\{\Delta H(Q)\}_{\mathcal{B}(x)}}{\text{vol}(\mathcal{B}(x))} \quad (3)$$

When $\mathcal{B}(x)$ is sensed, the sensed information affects the C-space entropy via each configuration q . $\text{igd}_q(x)$ is then the marginal contribution to information gain density via configuration q , if a point $x \in \mathcal{P}_u$ were to be sensed. Furthermore, $\text{igd}_q(x)$ equals 0 when $\mathcal{A}_u(q)$ does not contain x . So we need only compute the above summation over those q 's such that $x \in \mathcal{A}_u(q)$, also called the unknown C-zone of x [2,3], and denoted by $\chi_u(x)$. Intuitively, C-zone of x is the set of configurations such that the robot when placed in such a configuration contains point x . One could think of this as a generalization of inverse kinematics that applies to the entire robot body.³ Therefore one can write:

$$\widetilde{IGD}_{\mathcal{C}}(x) = \sum_{q \in \chi_u(x)} \text{igd}_q(x) \quad (4)$$

In [1,2], we showed that

$$\text{igd}_q(x) = \begin{cases} \lambda \cdot H(Q) - \lambda \cdot p(q) \cdot \frac{dH(Q)}{dp} & x \in \mathcal{A}_u(q) \\ 0 & \text{otherwise} \end{cases}$$

The above expression consists of two terms that contribute to expected entropy reduction per unit volume. The first term, $\lambda \cdot H(Q)$ is the expected contribution when $\mathcal{B}(x)$ contains an obstacle. Note that in this case, for each such event, the entropy would reduce to zero, since robot in configuration q would now be in collision. The change in entropy is therefore $H(Q)$, the

³ [14] used a similar concept but in a different context. Their aim was to come up with a C-space representation that is easy to modify.

entropy before scanning. The scaling factor λ simply reflects the probability (per unit volume) of the event that $\mathcal{B}(x)$ contains an obstacle under Poisson model. The second term, $\lambda \cdot p(q) \cdot \frac{dH(Q)}{dp}$ is the contribution when $\mathcal{B}(x)$ changes to free. In this event, the absolute amount of entropy reduction is zero almost everywhere, however, the derivative w.r.t. $vol(\mathcal{B}(x))$ is non-zero and is equal to $\lambda \cdot p(q) \cdot \frac{dH}{dp}$. The probability of this event under Poisson model is very high and approaches unity in the limit.

Substituting for $H(Q)$ and its derivative (differentiating expression 2 w.r.t. $p(q)$), the final expression for igd_q is

$$igd_q(x) = \begin{cases} -\lambda \log(1 - p(q)) & x \in \mathcal{A}_u(q) \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Combining Eq. (4) and (5), we get

$$\widetilde{IGD}(x) = \sum_{q \in \mathcal{X}_u(x)} -\lambda \cdot \log(1 - p(q)) \quad (6)$$

This expression, combined with the expression for void probability $p(q)$ in Eq. (1) completely determines the IGDF for a point FOV sensor assuming a Poisson point process for obstacle distribution. The point to be scanned next is the one that maximizes $\widetilde{IGD}_c(x)$.

The closed form expression for expected entropy reduction for a point FOV sensor assumes that there are no occlusion constraints, i.e., a point to be sensed will always result in a free/obstacle status. We now discuss the beam FOV sensor that models the occlusion constraints in the expected entropy formulation.

3.3 Beam Sensor Model: Occlusion Constraint

The beam sensor, as the name implies, senses along a beam (ray) of finite length, L , emanating from the sensor origin (See Figure 2). It returns the distance of the first hit point (obstacle) along the beam. Points along the beam that are in front of the hit point (i.e., closer to the origin than the hit point) are in free space. Points along the beam behind the hit point (i.e., farther from origin than the hit point) are deemed to be un-sensible (by occlusion constraints) in this particular sensing action and their status (obstacle/free/unknown) remains the same as it was before the sensing action. Therefore, for a particular sensing action, a point along the beam may acquire one of three possible states: 0 (free), 1 (obstacle) or u (unsensible).

For this more general case, when a region $\mathcal{V}(s)$ is being sensed, the IGD function is now defined over the space of all sensor configurations. Note that since the volume of the sensor FOV is still zero, it is the density function that is still relevant. For each sensor configuration, s , the IGD function assigns a real value that corresponds to the expected information gain per unit volume

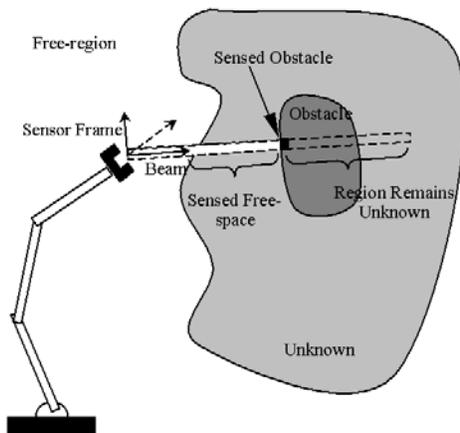


Fig. 2. The beam sensor model.

of sensor FOV with the sensor placed at configuration s . Using the same simplifications as for a point sensor (i.e., ignoring mutual entropy terms),

$$\widetilde{IGD}_C(s) = \lim_{\text{vol}(\mathcal{V}(s)) \rightarrow 0} \frac{E\{\Delta \tilde{H}(C)\}}{\text{vol}(\mathcal{V}(s))} = \sum_{q \in \chi_u(s)} \text{igd}_q(s)$$

and

$$\text{igd}_q(s) = \lim_{\text{vol}(\mathcal{V}(s)) \rightarrow 0} \frac{-E\{\Delta H(Q)\}}{\text{vol}(\mathcal{V}(s))} \quad (7)$$

As before, $\text{igd}_q(s)$ is the marginal contribution to IGD via configuration q when the sensor senses at configuration s . Similar to the point FOV sensor, the summation is restricted to the unknown C-zone of $\mathcal{V}_u(s)$ (defined in the next paragraph), denoted by $\chi_u(s)$, and defined as the set of q 's such that $\mathcal{A}_u(q) \cap \mathcal{V}_u(s) \neq \phi$.⁴

For mathematical formulation, we will assume that the sensor FOV, $\mathcal{V}(s)$, is a thin cylinder of infinitesimal radius (or equivalently, infinitesimal cross sectional area denoted by Δa) and length L . We discretize this cylinder into n “disks”, each of length Δl , by planes orthogonal to its axis, as shown in Figure 3. As Δa and Δl approach zero, the cylinder becomes an ideal beam.

Let $\mathcal{V}_u(s)$ ⁵ denote the portion of sensor FOV that lies inside \mathcal{P}_{unk} and is *in front of the first known obstacle* along the sensing direction, i.e., $\mathcal{V}_u(s)$ denotes the largest possible sensing region the beam sensor can sense at

⁴ This extends the notion of C-zone to that of a set in physical space rather than a single point x .

⁵ There is a slight abuse of notation for simplicity. Subscript u in $\mathcal{V}_u(s)$ denotes not only unknown part of $\mathcal{V}(s)$ but also further excludes those portions of unknown environment in $\mathcal{V}(s)$ that are occluded by known obstacles.

s . $\mathcal{V}_u(s) \approx x_1 \cup x_2 \cup \dots \cup x_m$ where $x_i, i = 1, \dots, m$ are the disks lying inside $\mathcal{V}_u(s)$ (we are concerned with only the unknown portion of sensor FOV and label only those disks). Note that $\mathcal{V}_u(s)$ may be a multiply connected set. If a disk (say, x_i) contains the hit point, its status would become 1 (obstacle). All disks $x_j, j < i$, would become 0 (free), and all disks $x_k, k > i$ would keep their status u (unknown) as shown in Figure 3. So we will get a “0, 0, ..., 0, 1, u, u, \dots, u ” sequence.

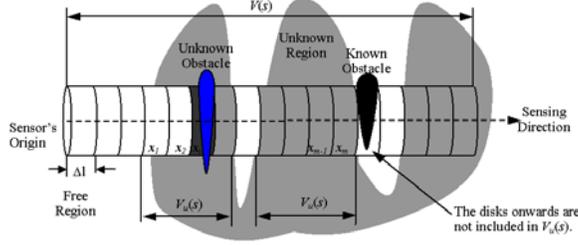


Fig. 3. The beam sensor model with infinitesimal width and discretized into “disks”. The hit point lies in disk x_i with $i = 3$. Disks x_1 and x_2 become free and disks x_4 onwards remain unknown.

The location of the hit point is a random variable. The event that the i^{th} disk contains the hit point, denoted by $x_i = h$, corresponds to first $(i - 1)$ disks, x_1, x_2, \dots, x_{i-1} , being all in \mathcal{P}_{free} and the i^{th} disk, x_i , containing an obstacle point, i.e., $\{x_j = 0, j = 1, \dots, i - 1 \wedge x_i = 1\}$ where $i \in \{1, \dots, m\}$. The corresponding probability, denoted by $p(x_i = h)$ is then given by using Eq. (1):

$$p(x_i = h) = e^{-\lambda \cdot (i-1) \cdot \Delta a \cdot \Delta l} \cdot (1 - e^{-\lambda \cdot \Delta a \cdot \Delta l}) \quad (8)$$

with $1 \leq i \leq m$, and $\Delta a \cdot \Delta l$ being the volume of each disk.

3.3.1 Compute $igd_q(s)$

The numerator (expectation) in $igd_q(s)$ in Eq. (7) is:

$$\begin{aligned} E\{\Delta H(Q)\}_s &= E\{\Delta H\}_{x_i \in \mathcal{A}(q)} + E\{\Delta H\}_{x_i \notin \mathcal{A}(q)} + E\{\Delta H\}_{\exists \text{ no hit pt}} \\ \Leftrightarrow igd_q(s) &= (igd_q)_1 + (igd_q)_2 + (igd_q)_3 \end{aligned} \quad (9)$$

The r.h.s. in the above expression consists of three terms: the first one corresponds to an x_i being a hit point and belonging to $\mathcal{A}(q)$ for the given q (event 1); the second term corresponds to an x_i being a hit point but not belonging to $\mathcal{A}(q)$ for the given q (event 2); the last term corresponds to there being no hit point in the sensed beam, i.e., all x_i 's are sensed free (event 3). The three cases are shown schematically in Figure 4. The robot (solid line) is holding the sensor at a given sensor configuration s . For a given q , the region

occupied by the robot, $\mathcal{A}(q)$ is shown in dotted. The hit point may lie inside $\mathcal{A}(q)$, outside $\mathcal{A}(q)$, or there may not be any hit point, i.e., the entire beam is free.

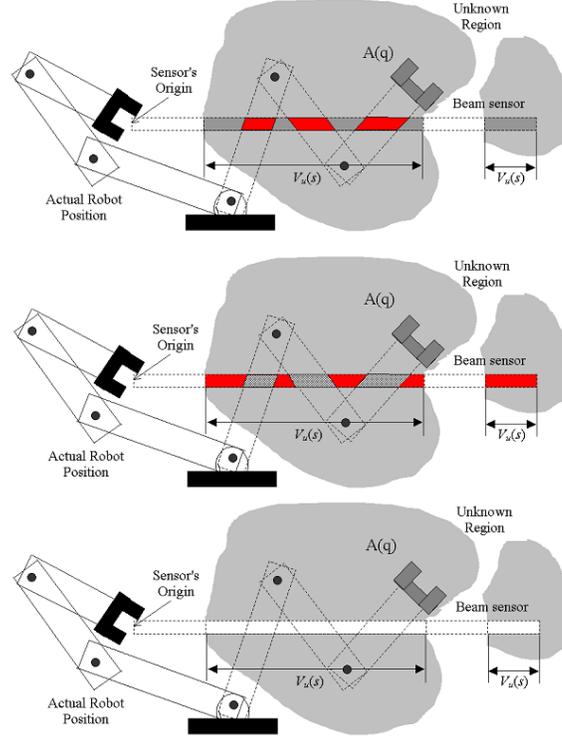


Fig. 4. Computation of $igd_q(s)$. The three terms on the r.h.s. in equation 9 correspond to the above three events. (top) When hit point is inside $\mathcal{A}(q)$. The hit point may lie anywhere inside the dark strip. (middle) When hit point is outside $\mathcal{A}(q)$. The hit point may lie anywhere inside the dark strip. (bottom) There is no hit point, i.e., the entire beam is sensed free.

Computing the 1st Component ($E\{\Delta H_1\}$) Derivation omitted.

When the hit point lies inside $\mathcal{A}(q)$, we have

$$(igd_q)_1 = \frac{\lambda}{L} \cdot len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot H(Q) \quad (10)$$

The expression makes intuitive sense. Since it is the expected contribution from those cases where a sensed hit point lies inside $\mathcal{A}(q)$, we know that in each such outcome, the entropy would reduce to zero since the status of Q would become known (in collision) and hence the entropy reduction will be $H(Q)$, the entropy before sensing. The multiplying factor $\frac{\lambda}{L} \cdot len(\mathcal{A}(q) \cap$

$\mathcal{V}_u(s)$) simply represents the expectation (per unit length) of such an event happening under Poisson model.

Computing the 2nd Component ($E\{\Delta H_2\}$) Derivation Omitted.

When the hit point lies outside $\mathcal{A}(q)$, we have

$$(igd_q)_2 = 0 \quad (11)$$

This implies for outcomes comprising event 2 (the hit point is sensed but does not lie within $\mathcal{A}(q)$ for a given q), the marginal information gain density due to configuration q is zero. We can think of $(igd_q)_2$ as essentially a summation (over outcomes comprising event 2) of “derivatives” of entropy w.r.t. volume weighted by the corresponding probability of such an outcome happening. The derivative is finite but the probability of the outcome happening (a hit point lying outside the robot but within the sensing beam) is very low and approaches zero under Poisson model as Δa approaches zero. $(igd_q)_2$ being a summation of these product terms, is also zero.

Computing the 3rd Component ($E\{\Delta H_3\}$) Derivation omitted.

When the whole beam is sensed free of hit points, we have

$$(igd_q)_3 = -\frac{\lambda}{L} \cdot len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot p(q) \cdot \frac{dH(Q)}{dp} \quad (12)$$

$(igd_q)_3$ is essentially the “derivative” of entropy w.r.t. volume weighted by the probability of the event 3 (when the sensor does not sense any hit point). The derivative is again finite, however, the probability event 3 is very high and approaches unity as Δa approaches zero under Poisson model. The $(igd_q)_3$ being the product of the two, is therefore finite.

We can easily get $igd_q(s)$ from Eq. (10), (11) and (12),

$$igd_q(s) = \begin{cases} \frac{\lambda}{L} \cdot len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot (H(Q) - p(q) \cdot \frac{dH(Q)}{dp}) \\ = -\frac{\lambda}{L} \cdot len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot \log(1 - p(q)) & x \in \mathcal{A}_u(q) \\ 0 & \text{otherwise} \end{cases}$$

And finally,

$$\widetilde{IGD}_c(s) = -\frac{\lambda}{L} \cdot \sum_{q \in \mathcal{X}_u(s)} len(\mathcal{A}(q) \cap \mathcal{V}_u(s)) \cdot \log(1 - p(q))$$

Note that the beam sensor result is as if occlusion does not matter within entropy computation and the net result is effectively a “summation” of point sensors along the beam. Moreover, we can easily see that as L , the length of the sensing beam, goes to zero, this result agrees with the one obtained for the point FOV sensor, i.e., we have $\lim_{L \rightarrow 0} \frac{\mathcal{A}(q) \cap \mathcal{V}_u(s)}{L} = 1$, and therefore, $igd_q(s) = \lambda \cdot (H(Q) - p(q) \cdot \frac{dH(Q)}{dp})$, precisely the result we obtained in [1–3]⁶.

⁶ Please note that there is an algebraic error in earlier papers for the point FOV case. This is the corrected result.

3.4 Generic Range Sensor with Non-zero Volume FOV

We now consider the general case, that of a range sensor whose FOV has non-zero volume, i.e., $\mathcal{V}(s)$ is an open set in R^3 , and the the actual volume sensed is governed by occlusion constraints. Most commercially available range sensors that provide range images (such as the area scan laser range sensor used in SFU eye-in-hand system [5]) fall into this category. Fig. 5 shows a schematic diagram as the sensor senses an unknown region within its FOV. As before, let $\mathcal{V}_u(s)$ denote the portion of the FOV that intersects \mathcal{P}_u and is not occluded by known obstacles. Note that $\mathcal{V}_u(s)$ might be a multiply-connected set. In the figure, $\mathcal{V}_u(s)$ consists of regions A, B, C and D (region E is excluded from $\mathcal{V}_u(s)$ since it is occluded by a known obstacle). After sensing, regions A, B and C become free; region D remains unknown because it is occluded by the sensed obstacles (shown in dark). Of course, the sensor also provides the distances from the sensor's origin to the sensed obstacles.

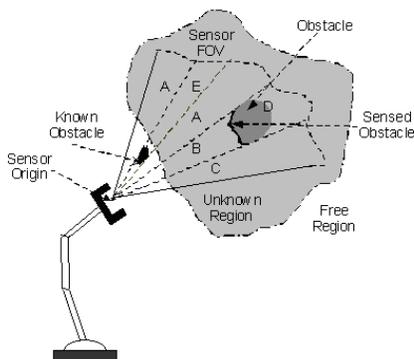


Fig. 5. Illustration of a generic range sensor's FOV $\mathcal{V}(s)$. After this sensing action, regions A, B and C become free, the black contour is a sensed obstacle and region D, occluded by the sensed obstacle remains unknown. Region E also remains unknown, but it is occluded by an already known obstacle.

Since the volume of the sensor is non-zero, clearly it is the information gain (IG) function (rather than the density) that is relevant. Making similar approximations as earlier,

$$\widetilde{IG}_C(s) = -E_s\{\Delta\widetilde{H}(C)\} = \sum_{q \in \mathcal{X}_u(s)} ig_q(s)$$

where $ig_q(s)$ is given by:

$$ig_q(s) = -E_s\{\Delta H(Q)\} \quad (13)$$

3.4.1 $IG_c(s)$ Computation

With occlusion constraint, the sensor can only detect the very first obstacle pt , called the hit point, along each sensing ray. Again, for a given q , $ig_q(s)$ is composed of two components, i.e., $ig_q = (ig_q)_1 + (ig_q)_2$.

The first component, denoted by $(ig_q)_1$, corresponds to those outcomes where the sensor would sense at least one hit point inside $\mathcal{A}(q) \cap \mathcal{V}_u(s)$, i.e., $hitpt \in \mathcal{A}(q) \cap \mathcal{V}_u(s)$, for a given configuration q . Let this set of outcomes be denoted event 1. After sensing, the robot, were it to be placed at configuration q , $\mathcal{A}(q)$, would be in collision with an obstacle (the sensed hit point). So $H(Q | \text{event 1}) = 0$ and $\Delta_{\text{event 1}} H(Q) = H(Q | \text{event 1}) - H(Q) = -H(Q)$.

It turns out (not unexpectedly in the light of beam sensor result) that the probability of event 1, $\Pr[hitpt \in \mathcal{A}(q) \cap \mathcal{V}_u(s)]$, is the same as $\Pr[pt \in \mathcal{A}(q) \cap \mathcal{V}_u(s)]$, as if occlusion does not matter! Theorem 1 states this result formally.

Theorem 1. $\Pr[hitpt \in \mathcal{B}] = \Pr[pt \in \mathcal{B}] = 1 - e^{-\lambda \cdot \text{vol}(\mathcal{B})}$ where $\mathcal{B} \subseteq \mathcal{V}_u(s)$ is any open set and pt are point obstacles whose distribution is governed by a Poisson point process.

Proof. Omitted. We outline the basic approach. We first discretize $\mathcal{V}(s)$ into N number of nearly-identical infinitesimal cones, $\mathcal{V}_1, \mathcal{V}_2, \dots, \mathcal{V}_N$, as shown in Fig. 6. Consider a cone (with apex at sensor origin) which contains $\mathcal{V}(s)$. Lay a discrete grid of size ϵ and connect the boundary of these cells to the sensor's origin. Thus, we get N infinitesimal cones. As $\epsilon \rightarrow 0$, $N \rightarrow \infty$, discretized sensor approaches original range sensor.

Now consider an open set $\mathcal{B} \subseteq \mathcal{V}_u(s)$. Indeed \mathcal{B} will intersect $M \leq N$ of these infinitesimal cones. Wlog, we label them $\mathcal{V}\mathcal{B}_1, \mathcal{V}\mathcal{B}_2, \dots, \mathcal{V}\mathcal{B}_M$. Also we denote the intersection of these $\mathcal{V}\mathcal{B}_i$'s with \mathcal{B} by $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_M$ respectively and use $front(\mathcal{B}_i)$ to denote the part of $\mathcal{V}\mathcal{B}_i$ that is in front of \mathcal{B}_i along the sensing direction. We use these infinitesimal cones to compute $\Pr[hitpt \in \mathcal{B}]$. Note that the occlusion for each pt in \mathcal{B}_i will happen in $front(\mathcal{B}_i)$. Again, since the obstacles are point obstacles, the probability of a pt being occluded approaches zero, although the actual proof is somewhat detailed and tedious and is omitted here for lack of space.

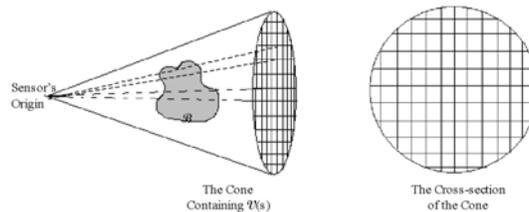


Fig. 6. The generic range sensor FOV discretized into N infinitesimal cones.

So we have

$$\begin{aligned}
(ig_q)_1 &= \Pr[\text{hitpt} \in \mathcal{A}(q) \cap \mathcal{V}_u(s)] \cdot H(Q) \\
&= \Pr[pt \in \mathcal{A}(q) \cap \mathcal{V}_u(s)] \cdot H(Q) \\
&= (1 - e^{-\lambda \cdot \text{vol}(\mathcal{A}(q) \cap \mathcal{V}_u(s))}) \cdot H(Q)
\end{aligned} \tag{14}$$

The second component, denoted by $(ig_q)_2$, corresponds to a set of outcomes in which there does not exist any hit point inside $\mathcal{A}(q) \cap \mathcal{V}_u(s)$. In this case, the status of $\mathcal{A}(q)$ would either remain unknown, albeit the unknown portion (volume) may have decreased, or it may become completely free; but it will not be known to be in collision. Let us denote this set of outcomes by event 2. Using the discretized FOV as in Figure 6, let us denote a possible state of $\mathcal{A}(q) \cap \mathcal{V}_u(s)$ after sensing by \underline{J} . Event 2 then corresponds to the set of outcomes $\{\underline{J} : \text{no hitpt} \in \mathcal{A}(q) \cap \mathcal{V}_u(s)\}$. By definition, then we have

$$(ig_q)_2 = \sum_{\underline{J} \in \text{event 2}} \Pr[\underline{J}] \cdot (H(Q) - H(Q | \underline{J})) \tag{15}$$

where $\Pr[\underline{J}]$ is the probability of $\mathcal{A}(q) \cap \mathcal{V}_u(s)$ being in state \underline{J} after sensing.

We show that the above expectation turns out to be that of the event (let us call it event 3) that there does not exist any pt in $\mathcal{A}(q) \cap \mathcal{V}_u(s)$, or equivalently that the region $\mathcal{A}(q) \cap \mathcal{V}_u(s)$ is free! This implies that occlusion does not matter in the expectation computation! It is not entirely unexpected in the light of beam sensor result. Theorem 2 states this result formally.

Theorem 2.

$$\begin{aligned}
&\sum_{\underline{J} \in \text{event 2}} \Pr[\underline{J}] \cdot H(Q | \underline{J}) = \Pr[\text{event 3}] \cdot H(Q | \text{event 3}) \\
&= e^{-\lambda \cdot \text{vol}(\mathcal{A}(q) \cap \mathcal{V}_u(s))} \cdot H(Q | \text{event 3})
\end{aligned} \tag{16}$$

Proof. Omitted.

Thus, expanding the summation in Eq. (15), we have,

$$(ig_q)_2 = H(Q) \cdot \sum_{\underline{J} \in \text{event 2}} \Pr[\underline{J}] - \sum_{\underline{J} \in \text{event 2}} \Pr[\underline{J}] \cdot H(Q | \underline{J}) \tag{17}$$

The first term (the first summation) above is $1 - \Pr[\text{hitpt} \in \mathcal{A}(q) \cap \mathcal{V}_u(s)]$. The expression for it is given by Theorem 1 if we substitute $\mathcal{A}(q) \cap \mathcal{V}_u(s)$ for \mathcal{B} . Furthermore, substituting from Theorem 2 for the second term, we get

$$\begin{aligned}
(ig_q)_2 &= H(Q) \cdot \Pr[\text{event 3}] - \Pr[\text{event 3}] \cdot H(Q | \text{event 3}) \\
&= e^{-\lambda \cdot \text{vol}(\mathcal{A}(q) \cap \mathcal{V}_u(s))} \cdot (H(Q) - H(Q | \text{event 3}))
\end{aligned} \tag{18}$$

So summing the two components together, we will have

$$ig_q = (ig_q)_1 + (ig_q)_2 = H(Q) - e^{-\lambda \cdot \text{vol}(\mathcal{A}(q) \cap \mathcal{V}_u(s))} \cdot H(Q | \text{event 3}) \tag{19}$$

Both $H(Q | \text{event 3})$ and $H(Q)$ in the above equation are determined using Eq. (2) and that $p(q | \text{event 3}) = e^{-\lambda \cdot \text{vol}(\mathcal{A}_u(q) \setminus \mathcal{V}_u(s))}$, where \setminus denotes set difference.

4 Algorithm for View Planning

Now that we have computed an expression for IG over sensor's configuration space, we can use the MER criterion to decide the next scan, i.e., choose the sensor configuration s_{max} such that $s_{max} = \max_s \{ \sum_{q \in \mathcal{X}_u(s)} ig_q(s) \}$. The algorithm

then is as follows:

```

for every s /* according to a certain resolution */
  determine  $\mathcal{V}_u(s)$ 
   $\widetilde{IG}(s) = 0$  /* initialize */
  for every q
    if ( $\mathcal{A}_u(q)$  overlaps with  $\mathcal{V}_u(s)$ )
      compute  $ig_q(s)$ 
       $\widetilde{IG}(s) = \widetilde{IG}(s) + ig_q(s)$ 
 $s_{max} = \max_s(\widetilde{IG}(s))$ 

```

Determining quantities such as $\mathcal{A}_u(q)$, $\mathcal{V}_u(s)$ involves straight forward geometrical computations⁷. For instance, determining $\mathcal{V}_u(s)$ corresponds to determining the intersection of the sensor FOV with \mathcal{P}_u while excluding portions of \mathcal{P}_u occluded by already known obstacles (before sensing action), a relatively simple geometric computation. $ig_q(s)$ for given s and q is therefore easily computed. The iteration over q , i.e., summation over C-space of the robot to determine $IG(s)$, may be prohibitive for robots with many degree of freedoms. In this case, the summation can be carried out over a large enough set of random samples [2]. The iteration over s , i.e., maximization over the sensor configuration space to determine s_{max} will be directly proportional to the number of discretized sensing configurations.

5 Simulation Results

In order to test the effectiveness of our formulae, we conducted a series of experiments on the simulated two-link eye-in-hand preliminary system shown in Figure 1. The task for the robot is to explore its environment, starting from its initial configuration. The overall planner used is SBIC-PRM (sensor-based incremental construction of probabilistic road map) reported in [3,5]. Briefly, SBIC-PRM consists of an incrementalized model-based PRM [15], that operates in the currently known environment; and a view planner that decides a reachable configuration within the currently known environment from which to take the next view. The two sub-planners operate in an interleaved manner. The simulation program, written in C++, runs on a Pentium III 800, and it takes about 53 seconds for each view planning iteration, corresponding to the algorithm in Section 4.

⁷ The representation of the robot and the physical space in this algorithm is similar to [5,8].

We now compare the results of four different view planning criteria for efficiency of exploration of the physical and configuration space. The first strategy, denoted by RV (random views), is to randomly choose a viewpoint as the next scan. The second strategy, denoted by MPV (maximum unknown physical volume) is to choose the next viewpoint so as to maximize the unknown physical volume inside the scan [6]. The third strategy is to use point FOV based MER criterion for view-planning [1,2], and place the centre of the actual FOV (the cone) at x_{max} , the single point that results in maximum entropy reduction. The fourth is to use the generic non-zero volume FOV based MER criterion derived in this paper. In all these cases, the robot started off as in Figure 1.

As shown in Figure 7, the first two strategies expand the known C-space much less than the last two MER criterion based strategy. Using RV gives us about 8% expansion of known C-space in 5 scans, and the robot reached its goal in 36 scans. MPV results in C-space expansion by about 54% in 5 scans. The point FOV based MER criterion gives us much better results, resulting in about 73% expansion in 5 scans. The general FOV based MER criterion was the best, better than point FOV based MER. It made the C-space expand by about 82% in 5 scans. For reader's information, although not relevant here, black dots in all these figures are the nodes of the probabilistic roadmap, built by SBIC-PRM planner and used for planning paths for the robot.

Figure 8 plots known C-space vs. number of iterations for the above four view-planning criteria. We can easily see that the generic range sensor based MER is the most efficient one, which expanded known C-space to about 90% in 7 scans; point FOV based MER needed 11 scans; RV needed 33 scans; and MPV needed 19 scans respectively.

6 Conclusions

We presented closed form solutions for computing the expected C-space entropy reduction for a general non-zero volume FOV range sensor while taking into account the occlusion constraints inherent in range sensors. This extends our previous results that applied to a point FOV sensor and did not account for occlusion constraints. Planar simulations show that our new results lead to more efficient exploration of the robot configuration space. As mentioned in the introduction, our next step is to implement these results for six-dof SFU eye-in-hand system, a PUMA 560 with a wrist mounted area scan laser range finder, reported in [5].

There are two simplifications in the current formulation. The first is that it assumes a Poisson point process for obstacle distribution that treats obstacles as points. Extending our formulation for a Boolean stochastic model [12] where geometric shape of obstacles is taken into account would be the next step. The second simplification is that we ignored mutual entropy terms for computational efficiency. Computation of these terms will involve com-

puting the intersection of robot's volume for pairs (and triplets, and so on) of unknown configurations. Even for second order mutual entropy terms, the computational complexity would be $O(N^2)$, where N is the number of robot configurations with the C-space discretized at a certain resolution. It will indeed be computationally expensive to calculate these terms [2]. The effect of these mutual entropy terms on the efficiency of exploration needs to be further investigated.

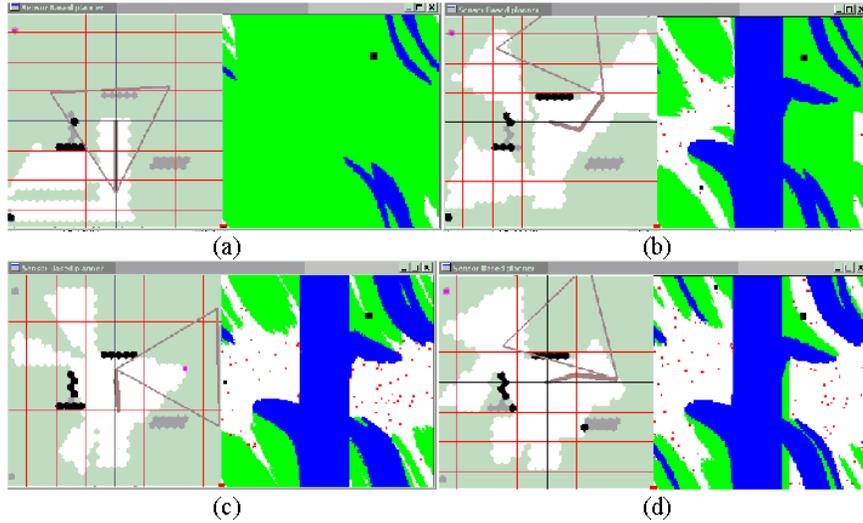


Fig. 7. Known Physical and C-space after 5 scans: (a) RV criterion, (b) MPV criterion, (c) Point FOV based MER criterion, (d) Generic non-zero volume FOV based MER criterion.

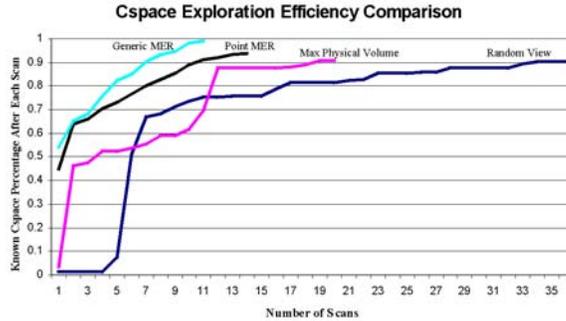


Fig. 8. The comparison of C-space exploration efficiency for the four view-planning algorithms: RV, MPV, Point FOV Based MER and Generic FOV Based MER

References

1. Y. Yu and K. Gupta. An Information Theoretic Approach to View Planning with Kinematic and Geometric Constraints. *Proc. IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, May 21-26, 2001, pp. 1948-1953.
2. Y. Yu. An Information Theoretical Incremental Approach to Sensor-based Motion Planning for Eye-in-Hand Systems. *Ph.D. Thesis*, School of Engineering Science, Simon Fraser University, 2000.
3. Y. Yu and K. Gupta. View Planning via C-Space Entropy for Efficient Exploration with Eye-in-hand Systems. *Proc. VII Int. Symp. on Experimental Robotics*, 2000. Available as lecture notes in *Control and Information Sciences*, LNCIS271, Springer. pp. 373-384.
4. Y. Yu and K. Gupta. Sensor-Based Motion Planning for Manipulator Arms: An Eye-in-Hand System. *Video Proc. IEEE Int. Conf. Robotics and Automation*, 2000.
5. Y. Yu and K. Gupta. Sensor-based Probabilistic Roadmaps: Experiments with an Eye-in-hand System. *Advance Robotics*, 2000, Vol.14, No.6, pp.515-537.
6. E. Kruse, R. Gutsche and F. Wahl. Effective Iterative Sensor Based 3-D map Building using Rating Functions in Configuration Space. *Proc. IEEE Int. Conf. on Robotics and Automation*. 1996, pp.1067-1072.
7. P. Renton, M. Greenspan, H. Elmaraghy, and H. Zghal. Plan-n-Scan: A Robotic System for Collision Free Autonomous Exploration and Workspace Mapping. *Journal of Intell. And Robotic Systems*, 1999. 24:207-234.
8. J. Ahuactzin and A. Portilla. A Basic Algorithm and Data Structure for Sensor-based Path Planning in Unknown Environment. *Proc. Of IROS 2000*, Vol. 2, pp. 903-908.
9. E. Cheung and V. J. Lumelsky. Motion Planning for Robot Arm manipulators with Proximity Sensors. *Proc. IEEE Int. Conf. on Robotics and Automation*, 1988, Vol. 2, pp. 740-745.
10. H. Choset and J. W. Burdick. Sensor based Planning for a Planar Rod Robot. *Proc. of the IEEE Int. Conf. on Robotics and Automation*, 1996, Vol. 4, pp. 3584-3591.
11. K. Hirai, M. Hirose, M. Haikawa and T. Takenaka. The Development of HONDA Humanoid Robot. *Proc. Of ICRA 1998*, pp. 1321-1326.
12. D. Stoyan and W.S. Kendall. *Stochastic Geometry and Its Applications*. J. Wiley, 1995.
13. H. G. Banos and J. C. Latombe. Robot Navigation for Automatic Construction using Safe regions. *Preprints Proc. ISER 2000*. pp. 395-404.
14. P. Leven and S. Hutchinson. Toward Real time Motion Planning in Changing Environments. *New Directions in Algorithmic and Computational Robotics (WAFR 2001)*. A. K. Peters, pp. 363-376.
15. L. Kavraki, P. Svestka, J. Latombe and M. Overmars, Probabilistic Roadmaps for Path Planning in High-dimensional Configuration Spaces, *IEEE Transactions on Robotics and Automation*, Aug. 1996, 12(4):556-580.