Autonomous Sensor Placement

Kevin H. Knuth[1,2] and Julian L. Center[2]
kknuth@albany.edu jcenter@ieee.org
1. University at Albany, Departments of Physics and Informatics, Albany NY USA
2. Autonomous Exploration, Inc., Andover MA USA

Abstract—*With an increasing reliance on robotic platforms to perform scientific exploration in remote or hostile environments, it is becoming crucial that robotic systems be able to perform autonomous intelligent sensor placement as well as autonomous experimental design. Such a system requires encoding of scientific knowledge, the ability to make inferences from data, and the ability to identify the most relevant question to ask given both the instrument's prior knowledge and the issue it is designed to address. This requires implementation of two computational engines: the inference engine and the inquiry engine. Here we demonstrate our first efforts to develop intelligent instruments that rely on autonomous sensor placement.*

1. Introduction

The scientific method relies on the cyclic application of three important steps: hypothesis generation, inquiry, and inference. We begin by forming a hypothesis or set of hypotheses. These hypotheses are examined via further inquiry by selecting and performing a relevant experiment. The results of the experiment are analyzed by inferring details of the hypothesis and by quantifying the remaining uncertainties. These results are used to modify the hypotheses and the learning cycle repeats.

Advances in the last decade in Bayesian inference have enabled us to automate the inferential process. More recently, careful application of information theory and the development of the inquiry calculus promise to enable the automation of inquiry [1]. In this paper we further describe our initial efforts to automate the process of inquiry [2].

The prototype is a robotic arm equipped with a light sensor. The robot is placed on a black field and is programmed to find a white circle or square in the field, and to correctly categorize it according to its center coordinates and radius, and in the case of a square, its orientation. The inference engine relies on a Bayesian Markov chain Monte Carlo (MCMC) algorithm, which not only estimates the model parameters, but also computes the evidence of the model. The inquiry engine relies on computing the entropy of a sampled set of probable measurements for each considered sensor placement position to decide which measurement or experiment is expected to provide the greatest amount of information. The inference-inquiry loop is then closed resulting in an instrument that takes measurements, makes inferences, and decides which measurements to take next. The robotic instrument stops when a pre-programmed level of tolerance is reached.

2. Information Gain and Inquiry

There has been great interest in automating the process of inquiry. These efforts have included cybernetics [3, 4], experimental design [5, 6, 7, 8, 9], active learning [10], sensor placement [11, 12, 13, 14], and inquiry [15, 16]. The general idea is that the potentially most productive data gathering activity is the one that maximizes information to be gained.

We begin by reviewing the theory from the perspective of Bayesian adaptive exploration [9]. Consider a proposed experiment E, which corresponds to taking a measurement at position (x_e, y_e). While we do not know for certain what we will measure, we can make a prediction based on the current posterior probability of the model $\mathbf{M}$. To accomplish this we write the probability of the measurement d_e in terms of the joint probability of d_e and $\mathbf{M}$ as

$$p(d_e \mid \mathbf{D}, (x_e, y_e), I) \equiv \int d\mathbf{M}\; p(d_e, \mathbf{M} \mid \mathbf{D}, (x_e, y_e), I) . \tag{9}$$

Using the product rule, we can write

$$p(d_e \mid \mathbf{D}, (x_e, y_e), I) \equiv \int d\mathbf{M}\; p(d_e \mid \mathbf{M}, \mathbf{D}, (x_e, y_e), I)\; p(\mathbf{M} \mid \mathbf{D}, (x_e, y_e), I) \tag{10}$$

This can be simplified by observing that, if we knew the precise values of the model parameters $\mathbf{M}$, we would not need the data $\mathbf{D}$

$$p(d_e \mid \mathbf{D},(x_e,y_e),I) \equiv \int d\mathbf{C}\; p(d_e \mid \mathbf{M},(x_e,y_e),I)\; p(\mathbf{M} \mid \mathbf{D},(x_e,y_e),I) \quad (11)$$

Probability theory is not sufficient to make a decision. To select a particular measurement location, we maximize the expected utility according to an assigned utility function: *U*(outcome, action)

$$(\hat{x}_e,\hat{y}_e) \equiv \int dd_e\; p(d_e \mid \mathbf{D},(x_e,y_e),I)\, U(d_e,(x_e,y_e)) \;, \quad (12)$$

where the measurement location (x_e,y_e) is indicative of the action and the measurement d_e is the predicted outcome. To select a measurement that provides the greatest expected gain in information, we use a utility function based on the information provided by the measurement. Using the Shannon information for our utility function we find

$$U(d_e,(x_e,y_e)) = \int d\mathbf{M}\; p(\mathbf{M} \mid d_e,\mathbf{D},(x_e,y_e),I)\log p(\mathbf{M} \mid d_e,\mathbf{D},(x_e,y_e),I) \quad (13)$$

By writing the joint entropy for **M** and d_e, and writing the integral two ways, one can show [9] that the optimal experiment can be found by maximizing the entropy of the possible measurements

$$(\hat{x}_e,\hat{y}_e) \equiv \arg\max_{(x_e,y_e)}\left(-\int dd_e\; p(d_e \mid \mathbf{D},(x_e,y_e),I)\log p(d_e \mid \mathbf{D},(x_e,y_e),I)\right) \quad (14)$$

Other utility functions could be used that depend on the energy required, the time it takes for the measurement, etc. These considerations will be important in a fully-functioning automated instrument.

This notion of maximizing the expected information gain promises to return the greatest amount of information on average in the case where one is interested in learning the values of all of the parameters of the model **M**. In many problems the model consists of a set of model parameters that are necessary to make predictions, but are in general, uninteresting. These model parameters are called nuisance parameters [16]. By simply maximizing the expected information gain with respect to the model, one risks selecting actions that will provide information about the nuisance parameters, but not the parameters of interest. To handle these situations rigorously, one must utilize the inquiry calculus to compute the relevance of questions.

A question can be defined as a set of possible statements [15, 1, 18, 19, 20]. Where a statement describes one's state of knowledge, a question describes a set of potential states of knowledge. Statements can be ordered according to whether one statement implies another, and this ordering can be generalized by defining a valuation quantifying the degree to which one statement implies another. This valuation on the space of statements is probability [18, 19, 20]. Similarly, questions can be ordered according to whether one question answers another. This ordering can again be generalized by defining a valuation that quantifies the degree to which one question answers another. This valuation on the space of questions, called relevance, also follows the sum rule, the product rule and the Bayes' theorem better known from probability theory [18, 19]. The relevance of a question must be related to the probabilities of its potential answers, which can be used to show that the relevance of a question that neatly partitions the possible answers can be quantified by the Shannon entropy of probability of those answers [18, 19, 20]. This inquiry calculus not only includes standard information theory, but extends it in a most useful way.

When one places a sensor and takes a measurement, one asks a question. In the event that there exists an array of possible measurement locations, each potential sensor location is associated with a particular question. The relevance of each of these questions can be computed using the inquiry calculus. The relevances can be compared to identify the most relevant question in the set. In the experiment we describe below, the most relevant measurement location is given by the entropy, just as expected from the arguments relying on information gain. However, we have conceived of more complex experiments, to be discussed elsewhere, that produce questions with relevances related to mutual information and other related quantities.

3. THE EXPERIMENT

Selecting a challenging experiment that simultaneously enables us to demonstrate these new computational technologies while avoiding the irrelevant details of a real-life problem requires that we remain restricted to a toy problem. Our problem of choice consists of locating and characterizing a white circle on a black field. While such characterization can trivially be accomplished using a camera system, we instead choose to perform this characterization using a simple point light sensor. This will require the instrument to take several light intensity measurements at a variety of locations. Such experiments are not without precedent. For example, surveying a geologic scene using Raman spectroscopy, which relies on excitation with a laser beam, naturally restricts one to a series of point measurements. Here we demonstrate that the

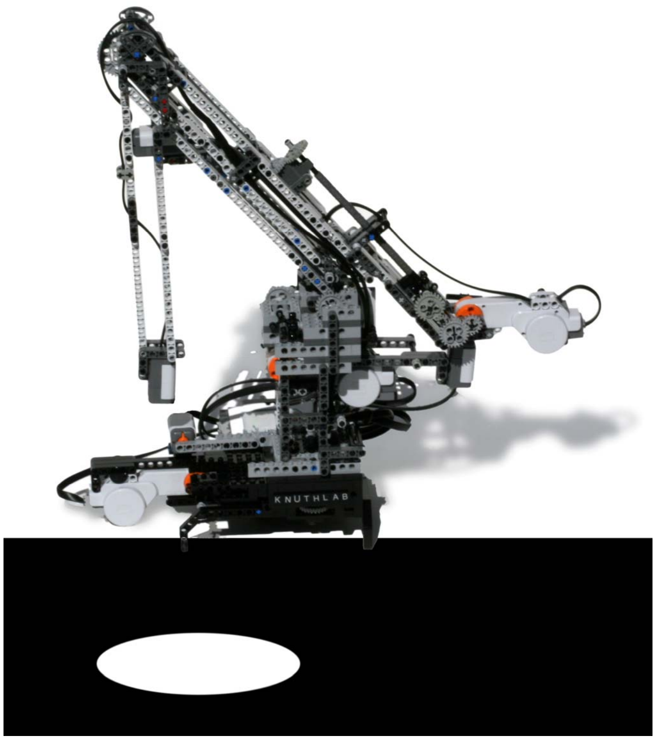

Figure 1 – The robotic arm and the playing field. The robot is equipped with a light intensity sensor and is programmed to search for and characterize the center location and radius of the white circle.

measurement locations can be selected intelligently on the fly by using this inquiry-based framework.

The instrument itself is a robotic arm holding a light intensity sensor. It was built using the LEGO Mindstorms NXT system (Figure 1). The arm has three degrees of freedom: rotation about the vertical axis (z-axis), and at two points about the y-axis (elbow and wrist). This mobility gives the arm access to an annular region of the horizontal plane. The light sensor, which is mounted at the end of the arm, is constrained to point vertically downward to ensure that the intensity measurements were taken as consistently as possible.

The robot is controlled by software running on a nearby Dell Latitude D610 laptop computer. It is programmed in MATLAB and operates in the Windows XP operating system. The laptop computer communicates with the robot via a Bluetooth Wireless connection to the LEGO Brick. The laptop interacts with the Brick by reading files, writing files and starting programs on the Brick. To request a measurement at a specified location, the MATLAB software must compute the number of motor rotations for each motor and write these values to a file on the LEGO Brick. MATLAB commands the motor program on the Brick to run, and this program reads the file and implements the instructions. When the robot is finished it creates a file containing the resulting light intensity value. To obtain this data, MATLAB reads this file begins its analysis and evaluation.

The LEGO Mindstorms NXT Brick is the computer that directly controls the motors and sensors of the robot. The Brick, which was originally developed by the MIT Media Lab, is programmed in the NXT-G programming language, which is a variant of LabVIEW. It has been programmed with a straightforward program that moves the arm from the home position to a position on the plane and records the light intensity. After writing the measurement result to a file, the arm returns to the home position.

4. Inference and Inquiry Engines

To characterize a circle, we require a model consisting of three parameters: the center location (x_0, y_0) and radius r_0. We will refer to the set of model parameters as

$$\mathbf{C} = \{(x_0, y_0), r_0\}. \qquad (15)$$

The inference engine implements Bayesian learning. It is provided with the model of the circle as well as a prior probability on the circle parameters. The prior probability is uniform in each of the model parameters and incorporates cut-offs defined by the range of the playing field as well as a minimum and maximum radius for the circle.

The inference engine is algorithmically based on a Monte Carlo algorithm called nested sampling [17], which provides the evidence of the model for use in model testing. Inference proceeds by taking the recorded data and evolving a set of 50 circles sampled from the prior probability. At the conclusion, nested sampling provides the evidence for the model as well as a set of samples. Often we require more samples than are returned by the nested sampling algorithm. In these cases, we simply further diversify the samples by employing Metropolis Hastings Markov chain Monte Carlo (MCMC).

At each step, the software displays the current state of both the inference and inquiry procedures. These sampled circles are displayed as the blue circles in Figure 2 and indicate probable circle locations and dimensions given the previously recorded light intensity levels. These samples are passed directly to the inquiry engine as a representation of the current posterior probability.

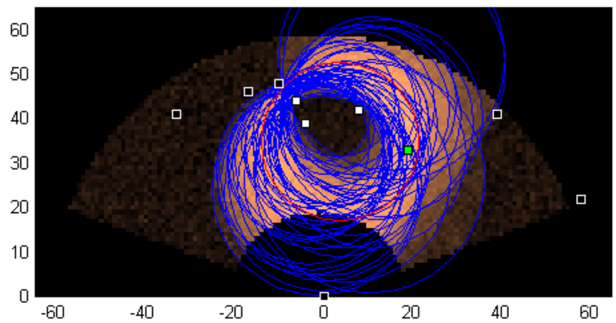

Figure 2 – The playing field with the inference and inquiry status overlaid. The coordinate axes are in the robot's native units. The semi-annulus reflects the range of reach of the robotic arm. White and black squares indicate relatively light and dark measurements, respectively. The blue circles indicate the set of circles sampled from the posterior probability. The red circle indicates the mean. The copper-toned shading denotes the entropy values for each of the possible measurement locations. Note that the algorithm finds that measurements in regions where the circle has been ruled out are less relevant. In addition, measurements in regions where the circle probably resides are also less relevant. The green square in the high entropy region of the map indicates the future measurement location. Notice that due to the nature of entropy, this measurement stands to rule out one-half of the potential circle positions and sizes. This results in an efficient binary search.

In this particular problem, the most relevant measurement location is computed using the entropy. The inquiry engine cycles through a grid of discrete measurement location candidates. For a given location candidate, each of the 50 sampled circles is queried as to what light intensity value is predicted to be observed. If the considered measurement location is positioned within the sampled circle, a measurement value is sampled from a Gaussian distribution with a mean value equal to the measurement intensity expected from a white background and an experimentally estimated standard deviation. If the measurement location lies outside of the sampled circle, the inquiry engine samples from a Gaussian distribution with a dark mean value and standard deviation. The set of predicted measurement results are binned and the entropy is computed. The measurement location with the greatest entropy is selected as the next sensor position.

In this demonstration, the robot performs an exhaustive search. This enables us to generate an entropy map, which can be seen in Figures 2 and 3 as the shaded copper background. Lighter background values reflect higher entropy indicating highly relevant measurement locations. Darker background values reflect lower entropy indicating less relevant measurement locations. The white and black squares indicate past light and dark measurement results. The green square situated firmly in the high entropy region denotes the next measurement location.

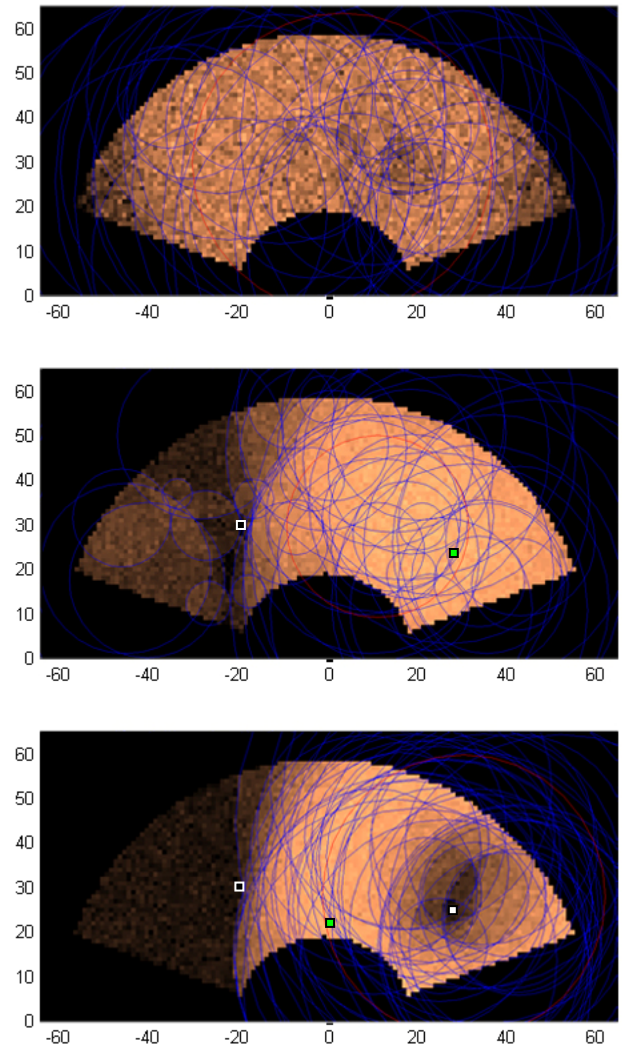

Figure 3 – The playing field at three different stages. (Top) With no data, the posterior samples are widely distributed and the corresponding entropy is essentially uniform. (Center) One dark measurement recorded at approximately (-20, 30) shifts the posterior samples to the right. (Bottom) A light measurement at the right focuses the posterior and makes nearby measurements less relevant.

5. RESULTS

The initial state of the robot is that of maximal uncertainty. The robot is programmed with a prior probability that encodes the information that the circle is present and within its reach, and has a radius that is bounded by a minimum and maximum value. The sampled circles in this case are essentially uniformly distributed, and the entropy map is more-or-less uniform indicating that all measurement locations are equally relevant (Figure 3, Top).

After a dark measurement is recorded, the posterior probability becomes biased indicating that the circle is probably not near that measurement point. The entropy map is naturally biased toward more distant unsearched areas of the playing field. Measurements near to the sampled point are less relevance than those far away (Figure 3, Center).

Once a point on the circle is identified, the posterior probability becomes more focused on a solution as indicated by the samples. Measurement locations near the point identified to be on the circle are less relevant as they too are probably on the circle (Figure 3, Bottom). The entropy map indicates that the most relevant measurements will serve to rule out one half of the circles in each step resulting in a maximally efficient binary search. This can be more clearly seen during the advanced stages of the experiment (Figure 2). On average, we have found the robot to take approximately 25 measurements to get to the desired accuracy of ± 0.5 distance units.

6. DISCUSSION

We present our initial efforts in designing intelligent science platforms, which automate the process of sensor placement. This investigation considers a straightforward experiment where the entropy in the distribution of predicted measurements is used to identify the most relevant measurement locations. We are currently working on experiments that involve model testing where the shape and number of the geometric objects to be characterized are unknown. In addition we are studying the effects of uncertainty in the robot's position on both its inferences and sensor location choices. In the future we will examine more complex experiments where the straightforward application of information gain as defined above is no longer applicable. In those cases we will rely on direct relevance calculations using the inquiry calculus.

8. REFERENCES

[1] K. H. Knuth, "The origin of probability and entropy," In: *Bayesian Inference and Maximum Entropy Methods in Science and Engineering, São Paulo, Brazil, 2008*, AIP Conf. Proc, Melville NY: American Institute of Physics, in press, 2008.

[2] K. H. Knuth, P. M. Erner, S. Frasso. "Designing intelligent instruments," In: K.H. Knuth, A. Caticha, J.L. Center, A. Giffin, C.C. Rodriguez (eds.), *Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Saratoga Springs, NY, USA, 2007*, AIP Conf. Proc. 954, Melville NY: American Institute of Physics, 203-211, 2007.

[3] N. Wiener, *Cybernetics or Control and Communication in the Animal and the Machine*, Cambridge: MIT Press, 1948.

[4] R. L. Fry, "The engineering of cybernetic systems," In: R.L. Fry (ed.) *Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Baltimore MD, USA*, AIP Conf. Proc. 617, Melville NY:AIP, pp. 497–528, 2002.

[5] V. V. Fedorov, *Theory of Optimal Experiments*. New York:Academic, 1972.

[6] D. V. Lindley, "On the measure of information provided by an experiment," *Ann. Math. Statist*. **27**, 986–1005, 1956.

[7] M. C. Shewry and H. P. Wynn, "Maximum entropy sampling," *J App. Stats.*, 14:165–170, 1987.

[8] P. Sebastiani and H. P. Wynn, "Maximum entropy sampling and optimal Bayesian experimental design," *J. Roy. Stat. Soc. B*, **62**, 145–157, 2000.

[9] T. J. Loredo, " Bayesian adaptive exploration," In: G. J. Erickson, Y. Zhai (eds.) *Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Jackson Hole WY, USA*, AIP Conf. Proc. 707, Melville NY:AIP, pp. 330–346, 2003.

[10] D. J. C. MacKay, "Information-based objective functions for active data selection," *Neural Computation*, **4**(4), 589–603, 1992.

[11] S.Thrun, W. Burgard, D. Fox, *Probabilistic Robotics*, MIT Press: Cambridge, 2005.

[12] A. Krause, A. Singh and C. Guestrin, "Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies," *JMLR*, 9:235–284, 2008.

[13] W. F. Caselton and J. V. Zidek, "Optimal monitoring network designs," *Statistics and Probability Letters*, 2(4):223–227, 1984.

[14] N. A. C. Cressie, *Statistics for Spatial Data*. Wiley, 1991.

[15] R. T. Cox, "Of inference and inquiry," In: R. D. Levine & M. Tribus (eds.) *The Maximum Entropy Formalism*, Cambridge:MIT Press, pp. 119–167, 1979.

[16] K. H. Knuth, "Intelligent machines in the 21st century: Automating the processes of inference and inquiry," *Phil. Trans. Roy. Soc. Lond. A*, Triennial Issue. **361**(1813):2859–73, 2003.

[17] D. S. Sivia and J. Skilling, *Data Analysis A Bayesian Tutorial*, (2nd Edition). Oxford Univ. Press: Oxford, 2006.

[18] K. H. Knuth, "What is a question?," In: C. Williams (ed.), *Bayesian Inference and Maximum Entropy Methods in Science and Engineering, Moscow ID 2002*, AIP Conf. Proc. 659, Melville NY:AIP, pp. 227–242, 2002.

[19] K. H. Knuth, "Lattice duality: The origin of probability and entropy," *Neurocomputing*. 67C: 245-274, 2005. DOI: 10.1016/j.neucom.2004.11.039

[20] K. H. Knuth, "Valuations on lattices and their application to information theory," (Invited paper), *Proceedings of the 2006 IEEE World Congress on Computational Intelligence (IEEE WCCI 2006)*, Vancouver, BC, Canada, July 2006, 2006.