Identification of control modes that maximizes goal disambiguation using Fisher Information

Context: The subject performs reaching tasks using an assistive robotic manipulator. The set of discrete goals is denoted by $\mathcal{G}$ with $n = |\mathcal{G}|$. Our aim is to identify those control dimensions/modes that will help the robot to disambiguate between the different goals and thereby help the robot perform better intent inference.
Intent inference: In order for the assistive robot to provide the right kind of assistance, it needs to have a good idea of what the user's underlying intentions are; or in other words, the robot needs to know which one of the n discrete goals is the user's intended goal. Furthermore, we assume that there is one and only one intended goal and is denoted by g^*.
Intent estimation: Intent inference essentially boils down to estimating $P(g|\boldsymbol{u_h}, \boldsymbol{x_r})$. The problem can be formally described in terms of POMDPs, in which a prior over goals and likelihood of user actions can be assumed. Bayesian inference techniques can be used to estimate the posterior distribution over goals to estimate $P(g|\boldsymbol{u_h}, \boldsymbol{x_r})$.

$$P(g|\boldsymbol{u_h}, \boldsymbol{x_r}) = \eta P(\boldsymbol{u_h}|g, \boldsymbol{x_r})P(g)$$

where η is a normalization factor.

Alternately, we could rely on heuristic methods based on *confidence functions* to estimate intent. Different types of confidence functions can be used. One choice of confidence function is a "directedness" based confidence function which measures the directedness of the user's control command towards a particular goal location. The higher the confidence associated with a goal, the more the system is confident that the goal is indeed the user's intended goal. Confidence functions can be interpreted as a "proxy" for $P(g|\boldsymbol{u_h}, \boldsymbol{x_r})$.

Another interpretation of a confidence function is that it is like a "virtual sensor" whose measurements contain information regarding the user's intended goal.
Control dimensions/modes: Due to the dimensionality mismatch between the robotic device and the control device used to control these devices, the entire control space is partitioned into smaller subsets called *control modes*.

In the most restricted cases (head arrays and sip n' puffs) the control modes correspond to the control dimensions themselves (1D interfaces). In such cases, the modes can be characterized by the control commands that can be generated along the corresponding control dimension.
Maximizing goal disambiguation: Given a particular form of intent estimation technique (Bayesian or confidence functions), certain control commands (or equivalently control dimensions) issued *by* the human will be *more* useful *for* the robot in performing goal disambiguation and therefore inferring the human's intended goal. Our aim is to develop a metric that will estimate the "goodness" of being in a control dimension (and as a result of issuing control commands along those control dimensions).

In our RSS 2017 work, we formulate a disambiguation metric that characterizes the disambiguation capability of each control dimension/mode. A mode switch assistance scheme is developed utilizing this metric in which the control mode with maximum disambiguation capability is chosen *for* the human upon assistance request. The hypothesis is that subsequent operation of the robot in the optimal control mode with "help" the robot to perform better intent inference and will result in appropriate kinds of assistance behavior which will in turn improve task related performance.

Fisher Information for characterizing control modes: The disambiguation metric developed in our previous work was completely ad hoc and heuristic. We hope to rely on more theoretically sound measures to characterize the disambiguation capability of a control mode. To this end, we rely on the Fisher Information metric.

Fisher information quantifies the ability of a random variable (in this case the confidence measure) to estimate an unknown parameter (which is the position of the intended goal). This idea is directly related to the approach taken by Silverman et al. and Miller et al., in which Fisher Information is used within the context of optimal control for information acquisition. In their work, the expected value of the Fisher Information with respect to the probability density function of the unknown parameter is computed (the positions of the intended goal) and is known as the *Expected Information Density* (EID). In our work we use EID to characterize the "information content" contained in a control mode in estimating the unknown parameter.

Scenario: We will restrict to a 2D reaching task for the sake of illustration.

The robot is a 2D robot that can perform reaching tasks. The control interface has 2 modes: corresponding to the x and y dimensions. The user can initiate motion along any dimension (one at a time). There are n discrete goals in the scene.

The set of goals are characterized by their position in space and is denoted by $\{\boldsymbol{x_{g_1}}, \boldsymbol{x_{g_2}}, \ldots, \boldsymbol{x_{g_n}}\}$. Let $\boldsymbol{x_r}$ denote the end-effector position with respect to the world frame and $\boldsymbol{u_h}$ denote the user control command. Let c_g denote the confidence measure associated with a goal g. We can interpret this confidence measure as a sensor reading.

Aim: Given all the goal locations, the prior distribution over goals and the robot position what we are interested in the *expected information density* of a control dimension. That is, how much information can be gained by moving in a particular control dimension regarding the location of the intended goal ($\boldsymbol{x_{g^*}}$).

Mathematical Formalism: The unknown parameter that we are trying to estimate is the location of the intended goal denoted by $\boldsymbol{x_{g^*}}$. The "measurement" model for the confidence "sensor" that captures the confidence of the intended goal, g^*, is given by

$$c_{g^*} = \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h}) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma^2)$$

We assume that our measurement model is stochastic. The noise is modeled as a zero-mean Gaussian distribution with a predefined variance. (Note that since this sensor is purely a mathematical construct and therefore the variance is artificially introduced into the picture by the system designer). Assuming a "directedness" based confidence formulation we have

$$\Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h}) = \frac{1 + cos(\theta)}{2} \quad \in [0, 1]$$

where

$$cos(\theta) = \frac{\boldsymbol{u_h} \cdot (\boldsymbol{x_{g^*}} - \boldsymbol{x_r})}{\|\boldsymbol{u_h}\| \cdot \|\boldsymbol{x_{g^*}} - \boldsymbol{x_r}\|}$$

The unknown parameter is the location of the intended goal g^* and this could be any one of the n goals from the set $\mathcal{G}$.

Since c_{g^*} is distributed as a Gaussian random variable (by definition), the Fisher information (the amount of information a "measurement" has regarding $\boldsymbol{x_{g^*}}$) is given by

$$\mathcal{I}_{i,j}(\boldsymbol{x_{g^*}}, \boldsymbol{u_h}, \boldsymbol{x_r}) = \frac{1}{\sigma^2} \frac{\partial^2 \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h})}{\partial x^i_{g^*} x^j_{g^*}}$$

Since $\boldsymbol{x_{g^*}}$ is a 2D parameter, the Fisher Information is a 2 by 2 matrix and not a scalar. Since the intended goal could be any one of the goals from $\mathcal{G}$ we take expectation over $\boldsymbol{x_{g^*}}$. Therefore,

$$\Phi_{i,j}(\boldsymbol{u_h}, \boldsymbol{x_r}) = \frac{1}{\sigma^2} \int_{x^i_{g^*}} \int_{x^j_{g^*}} \frac{\partial^2 \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h})}{\partial x^i_{g^*} x^j_{g^*}} p(x^i_{g^*}, x^j_{g^*}) dx^j_{g^*} dx^i_{g^*}$$

The priors over the goals are only defined at discrete points in the domain, since the goal locations themselves are discrete. Therefore,

$$p(\boldsymbol{x_{g^*}}) = \begin{cases} p_i & for \;\; \boldsymbol{x_{g^*}} = \boldsymbol{x_{g_i}} \\ 0 & otherwise \end{cases}$$

such that $\sum_{i=1}^{n} p_i = 1$.
Marginalizing over $\boldsymbol{x_r}$

$$\Phi_{i,j}(\boldsymbol{u_h}) = \int_{\boldsymbol{x_r}} \left[\frac{1}{\sigma^2} \int_{x^i_{g^*}} \int_{x^j_{g^*}} \frac{\partial^2 \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h})}{\partial x^i_{g^*} x^j_{g^*}} p(x^i_{g^*}, x^j_{g^*}) dx^j_{g^*} dx^i_{g^*} \right] p(\boldsymbol{x_r}) d\boldsymbol{x_r}$$

(The integral with respect to $\boldsymbol{x_r}$ is also a double integral, but is written vectorially due to space constraints.)

We assume perfect knowledge of where the robot is. Therefore the probability density function for $\boldsymbol{x_r}$ is given by

$$p(\boldsymbol{x_r}) = \begin{cases} 1 & \boldsymbol{x_r} = \boldsymbol{x_{true}} \\ 0 & otherwise \end{cases}$$

Therefore $\Phi_{i,j}(\boldsymbol{u_h})$ reduces to

$$\Phi_{i,j}(\boldsymbol{u_h}) = \frac{1}{\sigma^2} \int_{x^i_{g^*}} \int_{x^j_{g^*}} \frac{\partial^2 \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h})}{\partial x^i_{g^*} x^j_{g^*}} p(x^i_{g^*}, x^j_{g^*}) dx^j_{g^*} dx^i_{g^*} \Bigg|_{\boldsymbol{x_r} = \boldsymbol{x_{true}}}$$

The EID for $\boldsymbol{u_h}$ is then computed as

$$EID(\boldsymbol{u_h}) = \det(\bar{\Phi}(\boldsymbol{u_h}))$$

The EID for control command along x dimension can be computed by summing up $EID([1,0])$ and $EID([-1,0])$ and similarly for y dimension can be computed by the summation of $EID([0,1])$ and $EID([0,-1])$.

Simulation Results

Simulations were done in MATLAB to understand what kind of information $\Phi(\boldsymbol{u_h})$ encoded. The number of goals were varied from 2 to 4. EID was calculated for x and y dimensions.

The EID was consistently higher for that control dimension which would result in an increase of confidences for **ALL** goals. The system does not know which is the user's intended goal. It assigns a probability for each one of the n goals to be the intended goal (the prior). The EID is computed by taking the expectation over all possible intended goals. Higher EID will be for that control command (control dimension) which would result in a higher confidence regardless of which one of the n goals the user chose as the intended goal.

However, this is not exactly what we are looking for as this is not goal disambiguation. In order for the system to be able to disambiguate it should be able to identify dimensions along with confidence related to the true intended goal rises and all other confidences get suppressed.

This is probably an issue with the choice of measurement model. Should the "sensor" be sensitive to absolute values of the confidence or the "differences"?

Possible solutions and issues

Instead of using the confidence in the intended goal as the "virtual sensor" we could possibly use the negative of entropy of the confidence distribution as the sensor readings. By doing so, the control dimension which will maximize this quantity will have higher EID. I am still trying to work out the details for this now.

Issues: FIM requires the measurement model to be differentiable with respect to the unknown parameter which is the intended goal. If the "virtual sensor" is the negative entropy of the confidence distribution, it no longer is a function of the intended goal, but rather a function of the entire distribution. This will make the derivative ill-defined and subsequently we will not be able to evaluate the FIM.

Alternate idea

Can this problem be framed in a different way? What if the unknown parameter, is the control dimension which can disambiguate the most?

Consider the discrete set of control commands along the different axis. For a full 6DOF problem, this would be 12 different control commands corresponding to positive and negative directions of a control dimension?

Math Formalism for Entropy Based approach

Instead of

$$c_{g^*} = \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h}) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma^2)$$

we can possibly have

$$e = \Upsilon(\boldsymbol{x_r}, \boldsymbol{u_h}; \boldsymbol{x_{g_1}}, \ldots, \boldsymbol{x_{g_n}}) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma^2)$$

where Υ is the entropy (negative of) of the confidence distribution and is a function of robot position and user control command. The goal positions are given and are treated as fixed parameters. Individual confidences are given as

$$c_{g^i} = \frac{1 + cos(\theta)}{2}$$

where

$$cos(\theta) = \frac{\boldsymbol{u_h} \cdot (\boldsymbol{x_{g^i}} - \boldsymbol{x_r})}{\|\boldsymbol{u_h}\| \cdot \left\|\boldsymbol{x_{g^i}} - \boldsymbol{x_r}\right\|}$$

Normalizing the confidences across the goals will result in a "proxy" probability distributions over goal confidences.

$$p(g^i) = \eta c_{g^i} \quad , \eta = \Sigma_{i=1}^{n_g} c_{g^i}$$

η is the normalization constant.

Entropy of this distribution over goals is defined as

$$H(g) = -\Sigma_{i=1}^{n_g}\, p(g^i)\; log[p(g^i)]$$

and

$$\Upsilon(\boldsymbol{x_r}, \boldsymbol{u_h}; \boldsymbol{x_{g_1}}, \ldots, \boldsymbol{x_{g_n}}) = -H(g) = -(-\Sigma_{i=1}^{n_g}\, p(g^i)\; log[p(g^i)])$$

It can be seen that $H(g)$ is a function of $\boldsymbol{u_h}$ and $\boldsymbol{x_r}$ given the goal positions (treated as fixed parameters). Given a set of goals and the form of the confidence function, entropy is a scalar defined for each robot position and user control command.

I can possibly compute the Jacobian of Υ with respect to the robot position, however it will be different from how Murphey et al. conceives FIM. Murphey et al. has a clear notion of what is unknown (the goal position) and how some other variable (the sensor position) will help in knowing the unknown. (Fisher information is about how a sample of data contains information about an unknown parameter!)

Interpreting FI:

$$\mathcal{I}_{i,j}(\boldsymbol{x_{g^*}}, \boldsymbol{u_h}, \boldsymbol{x_r}) = \frac{1}{\sigma^2} \frac{\partial^2 \Upsilon(\boldsymbol{x_{g^*}}, \boldsymbol{x_r}, \boldsymbol{u_h})}{\partial x^i_{g^*} x^j_{g^*}}$$

The above equation is the Fisher information when the measurement was the confidence itself (earlier case). How do we interpret this quantity? It is the rate of change of the "measurement" (the confidence in this case) upon infinitesimal changes in the intended goal's position (the unknown parameter).

If this is the case, is it meaningful to have the "measurement" as the entropy of the confidence distribution and compute the derivative with respect to the robot position itself? For the given robot position $\boldsymbol{x_r}$, we want to be able to identify the $\boldsymbol{u_h}$ that will maximize the negative entropy of the confidence distributions, regardless of which one of the n_g goals is the true intended goal.

FI for Entropy Measurement:

Let $\boldsymbol{x_{g_1}}, \ldots, \boldsymbol{x_{g_n}}$ (the fixed parameters) be denoted as $\boldsymbol{\Theta}$

$$\mathcal{I}_{i,j}(\boldsymbol{x_r}, \boldsymbol{u_h}; \boldsymbol{\Theta}) = \frac{1}{\sigma^2} \frac{\partial^2 \Upsilon(\boldsymbol{x_r}, \boldsymbol{u_h}; \boldsymbol{\Theta})}{\partial x^i_r x^j_r}$$

Note: This is FI measure is not quite meaningful because the derivative is taken with respect to an 'known' parameter (the robot position). In Murphey et al.'s work, the FI is computed by taking the derivative with respect to the 'unknown' parameter; the location of the goal and the measurement function is truly the measurement model of the sensor that is being used. In our case the "virtual sensor" is the entropy "sensor". Taking the derivative of the entropy measurement with respect to the robot position is quite arbitrary and is motivated by the need to marginalize out $\boldsymbol{x_r}$.

Marginalizing over $\boldsymbol{x_r}$

$$\Phi_{i,j}(\boldsymbol{u_h};\boldsymbol{\Theta}) = \frac{1}{\sigma^2}\int_{x_r^i}\int_{x_r^j}\frac{\partial^2\Upsilon(\boldsymbol{x_r},\boldsymbol{u_h};\boldsymbol{\Theta})}{\partial x_r^i x_r^j}p(x_r^i, x_r^j)dx_r^j dx_r^i$$

Since the robot position is known perfectly the above integral reduces to a single evaluation at $\boldsymbol{x_r} = \boldsymbol{x_{true}}$.

Therefore

$$\Phi_{i,j}(\boldsymbol{u_h};\boldsymbol{\Theta}) = \left.\frac{1}{\sigma^2}\frac{\partial^2\Upsilon(\boldsymbol{x_r},\boldsymbol{u_h};\boldsymbol{\Theta})}{\partial x_r^i x_r^j}\right|_{\boldsymbol{x_r}=\boldsymbol{x_{true}}}$$

The expected information density for $\boldsymbol{u_h}$ is then given as

$$EID(\boldsymbol{u_h}) = det(\bar{\Phi}(\boldsymbol{u_h};\boldsymbol{\Theta}))$$

Entropy is a function of the given goal positions, robot position and the control commands. It does not matter which one of the n_g goals is the true intended goal, the entropy is completely determined by $\boldsymbol{u_h}, \boldsymbol{x_r}$ and $\boldsymbol{x_g}'s$. Certain control commands will result in a higher entropy of the confidence distribution and certain others will result in a lower entropy for the same robot position and goal configurations.

Our goal is to use Fisher Information as a metric to capture the "goodness" of moving along a particular control dimension (equivalent to $\boldsymbol{u_h}$) in disambiguating BETWEEN the goals. The "measurement" model cannot just be the confidences, since FI will be higher for that $\boldsymbol{u_h}$ that will result in a higher confidences for ALL goals. If the measurement model is able to capture the distribution of confidences (negative entropy for example), the FI will be higher when the entropy is maximized (or is it?).

Preliminary Results/Insights

I have an implementation of this newer idea and the results are not that promising. It might be because of the arbitrary decision to compute FI using derivative with respect to a known parameter which is the robot position. When using the entropy of the confidence distribution, it is unclear to me what is the unknown parameter. Is it the intended goal, is it the control command that will disambiguate the best?

Yet Another Formulation!

Recap: Going back to the problem addressed by Miller et al; the unknown parameter is the 2D spatial location of a goal. The "sensor" which measures the field potential can move around in the same space (space of 2D spatial location of a goal). Once the unknown parameter and the measurement model is defined, Fisher information can be used effectively to quantify the quality of a measurement in revealing the value of the unknown parameter.

It is imperative that if FI metric is to be used for identifying control modes with maximum disambiguation, the unknown parameter and the measurement model have to be identified properly. The upcoming formulation treats the control commands themselves as the unknown parameter and the (negative) entropy of the confidence distribution is the sensor model.

Casting the problem in the space of u_h

We restrict ourselves to the 2D scenario for ease of illustration. For convenience, we can assume that the control interface is 1D and can only generate unit magnitude control commands. A single point in the 2D space of $\boldsymbol{u_h}$ corresponds to a specific control command issued by the human. All points on the unit circle in this space is the set of all points whose magnitude is 1. The intersection of the unit circle and the x-y axes denote set of control commands available in control mode 1 and 2 respectively.

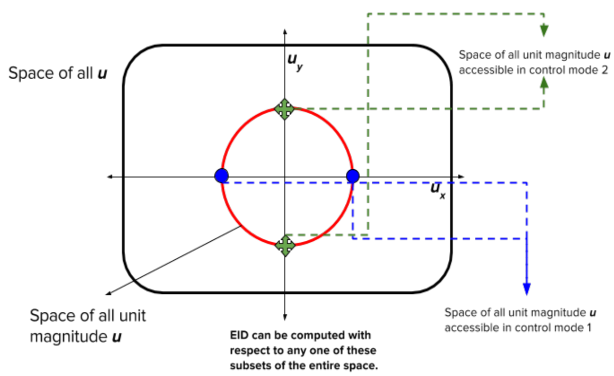

Figure 1: Illustration of subspaces of control command space.

Imagine that the virtual sensor that measures the "entropy of confidence distributions" is moving in the space of $\boldsymbol{u_h}$. In some parts of the space, the sensor readings will peak and in others they will not. This is akin to the electric field sensor getting close to the true spatial location of the goal in Miller et al.

Math Formalism

The entropy of the confidence distribution is the measurement model and can be written as,

$$e = \Upsilon(\boldsymbol{u_h}; \boldsymbol{x_r}, \boldsymbol{x_{g_1}}, \ldots, \boldsymbol{x_{g_n}}) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma^2)$$

Let $\boldsymbol{x_r}, \boldsymbol{x_{g_1}}, \ldots, \boldsymbol{x_{g_n}}$ be denoted by $\boldsymbol{\Theta}$. The above equation then simplifies to

$$e = \Upsilon(\boldsymbol{u_h}; \boldsymbol{\Theta}) + \delta, \quad \delta \sim \mathcal{N}(0, \sigma^2)$$

where Υ is the entropy (negative of) of the confidence distribution and is a function of robot position and user control command. The robot and goal positions are treated as fixed parameters. Individual confidences are given as

$$c_{g^i} = \frac{1 + cos(\theta)}{2}$$

where

$$cos(\theta) = \frac{\boldsymbol{u_h} \cdot (\boldsymbol{x_{g^i}} - \boldsymbol{x_r})}{\|\boldsymbol{u_h}\| \cdot \left\|\boldsymbol{x_{g^i}} - \boldsymbol{x_r}\right\|}$$

Normalizing the confidences across the goals will result in a "proxy" probability distributions over goal confidences.

$$p(g^i) = \eta c_{g^i} \quad, \eta = \Sigma_{i=1}^{n_g} c_{g^i}$$

η is the normalization constant.

Entropy of this distribution over goals is defined as

$$H(g) = -\Sigma_{i=1}^{n_g}\, p(g^i)\, log[p(g^i)]$$

and

$$\Upsilon(\boldsymbol{u_h}; \boldsymbol{\Theta}) = -H(g) = -(-\Sigma_{i=1}^{n_g}\, p(g^i)\, log[p(g^i)])$$

It can be seen that $H(g)$ is a function of $\boldsymbol{u_h}$, given the robot position and goal positions (treated as fixed parameters). Given a set of goals and the form of the confidence function, entropy is a scalar defined for each robot position and user control command.

Fisher Information Metric

$$\mathcal{I}_{i,j}(\boldsymbol{u_h}; \boldsymbol{\Theta}) = \frac{1}{\sigma^2} \frac{\partial^2 \Upsilon(\boldsymbol{u_h}; \boldsymbol{\Theta})}{\partial u_h^i u_h^j}$$

In order to compute the expected Fisher information density over different subspaces of the control command space we can compute the expectation of the above quantity over any subspace of interest $\mathcal{U}$. Each subspace (set of points) can correspond to a control mode. In the 2D case, with 1D control modes each mode contains two points; one corresponding to positive and negative unit magnitude control commands along the dimension of interest.

$$\Phi_{i,j}^{\mathcal{U}}(\boldsymbol{\Theta}) = \frac{1}{\sigma^2} \int_{u_h \in \mathcal{U}} \frac{\partial^2 \Upsilon(\boldsymbol{u_h}; \boldsymbol{\Theta})}{\partial u_h^i u_h^j} p(\boldsymbol{u_h}) d\boldsymbol{u_h}$$

Note that the above integral is a multidimensional integral, but has been written using a "vector" notation for convenience sake.

Illustration of computation with 1D interface

We will assume that only unit magnitude control commands are possible with the interface.
Control mode 1 ($\mathcal{U}_1$) corresponds to the set $[\{1,0\},\{-1,0\}]$ and control mode 2 ($\mathcal{U}_2$) corresponds to the set $[\{0,1\},\{0,-1\}]$.
Let $p(\{0,1\}) = p(\{0,-1\}) = p(\{1,0\}) = p(\{-1,0\}) = 0.5$.
Computation of $\Phi^{\mathcal{U}_1}_{i,j}$:

$$\Phi^{\mathcal{U}_1}_{i,j}(\mathbf{\Theta}) = \frac{1}{\sigma^2}\Big[\left.\frac{\partial^2\Upsilon(\boldsymbol{u_h};\mathbf{\Theta})}{\partial u_h^i u_h^j}\right|_{\boldsymbol{u_h}=\{1,0\}} p(\{1,0\}) + \left.\frac{\partial^2\Upsilon(\boldsymbol{u_h};\mathbf{\Theta})}{\partial u_h^i u_h^j}\right|_{\boldsymbol{u_h}=\{-1,0\}} p(\{-1,0\})\Big]$$

Computation of $\Phi^{\mathcal{U}_2}_{i,j}$:

$$\Phi^{\mathcal{U}_2}_{i,j}(\mathbf{\Theta}) = \frac{1}{\sigma^2}\Big[\left.\frac{\partial^2\Upsilon(\boldsymbol{u_h};\mathbf{\Theta})}{\partial u_h^i u_h^j}\right|_{\boldsymbol{u_h}=\{0,1\}} p(\{0,1\}) + \left.\frac{\partial^2\Upsilon(\boldsymbol{u_h};\mathbf{\Theta})}{\partial u_h^i u_h^j}\right|_{\boldsymbol{u_h}=\{0,-1\}} p(\{0,-1\})\Big]$$

EID of $\mathcal{U}_1$ can be computed as $\det(\bar{\Phi}^{\mathcal{U}_1}(\mathbf{\Theta}))$ and EID of $\mathcal{U}_2$ as $\det(\bar{\Phi}^{\mathcal{U}_2}(\mathbf{\Theta}))$ with $\mathbf{\Theta}$ to be the current robot position and the goal positions.

If EID of $\mathcal{U}_1$ is greater than EID of $\mathcal{U}_2$ it implies that control mode 1 has greater disambiguation capability.

Preliminary insights and issues

This formulation seems to work the best compared to all others. There are certain numerical issues (divide by zero et cetera) that are problematic and are dealt with by making approximations to the control commands. The approximations do have an impact on the results, but they are minor I believe.

Following are some of the "point cloud" plots which depicts the best control mode for different points in the workspace. Control mode 1 (red) controls the X axis, and mode 2 (blue) controls the Y Axis.

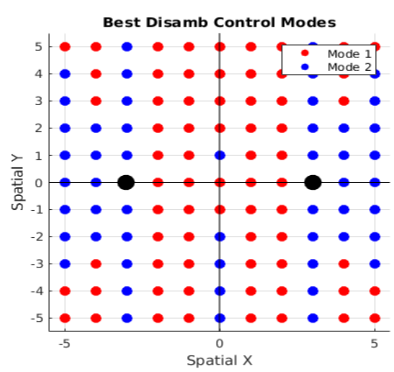

Figure 2: Disambiguating control modes.

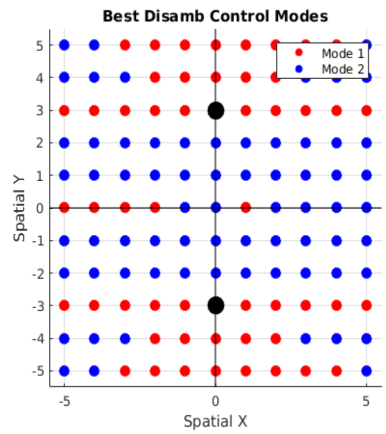

Figure 3: Disambiguating control modes.

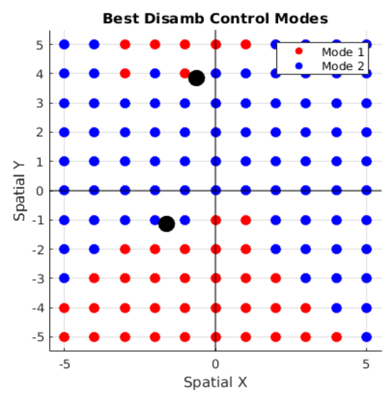

Figure 4: Disambiguating control modes.

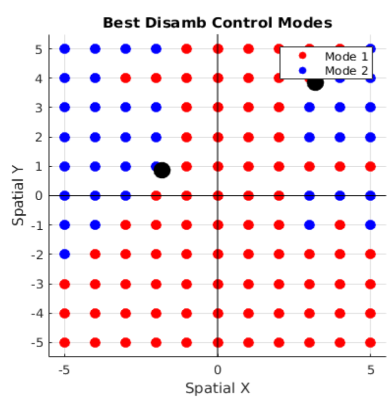

Figure 5: Disambiguating control modes.

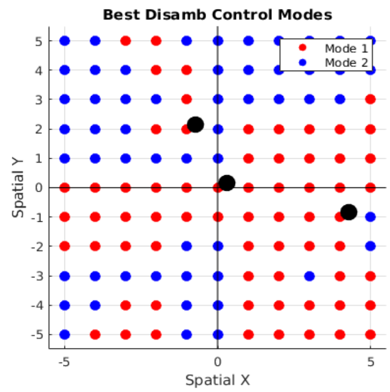

Figure 6: Disambiguating control modes.

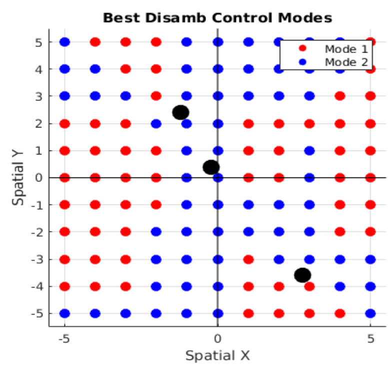

Figure 7: Disambiguating control modes.

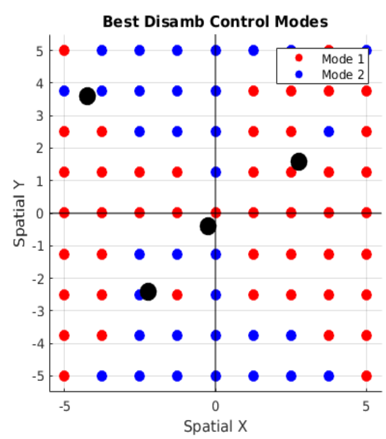

Figure 8: Disambiguating control modes.

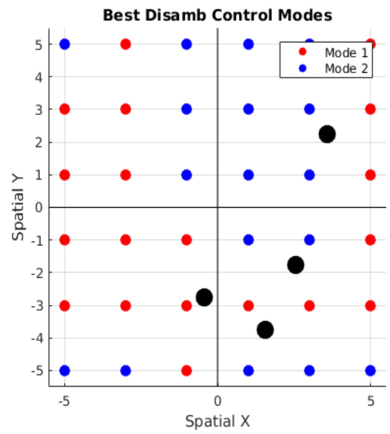

Figure 9: Disambiguating control modes.

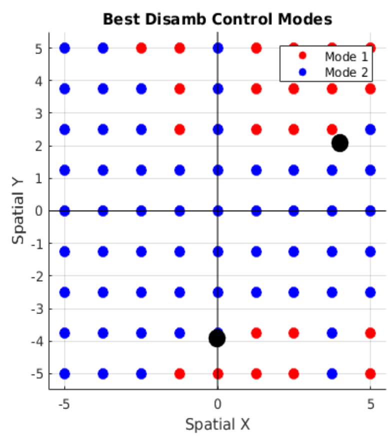

Figure 10: Disambiguating control modes.

Reflections

"...The Fisher Information $\mathcal{I}(\theta, x)$ can be thought of as the amount of information a measurement provides at location x for a given estimate of θ..."
"...Fisher information quantifies the ability of a random variable, in our case a measurement, to estimate an unknown parameter...

In the light of the above statements from Todd's papers, how justifiable is our approach!.

References

1. Silverman, Y., L. M. Miller, M. A. MacIver, and T. D. Murphey, "Optimal Planning for Information Acquisition", IROS 2013, 2013.
2. Miller, L. M., and T. D. Murphey, "Optimal Planning for Target Localization and

Coverage Using Range Sensing", IEEE Int. Conf. on Automation Science and Engineering (CASE): 2015, pp. 501-508, 2015.