Abstract

Current approaches to autonomous exploration focus on collecting observations in the absence of prior knowledge of the phenomena under investigation. However, it is unlikely that robots will arrive at planetary bodies without scientists having formed one or more hypotheses explaining data collected by precursor operations such as satellite images. These exploring robots collect observations to falsify the proposed hypotheses, incorporating those hypotheses can increase the efficiency of observation collection. This paper presents a novel algorithm, formulated in an exploration/exploitation framework, that directs robots to collect samples to determine which of a collection of hypotheses best explain data observed in situ by robots. We simulate a geologic exploration mission with a lander vehicle that can hop between locations of interest. This application is analogous to exploring of, e.g., the Aitken Basin of the south pole of Earth’s Moon where sampling sites need to be separated hundreds or thousands of meters. We demonstrate that sampling algorithms aware of the hypotheses under investigation perform statistically significantly better than standard approaches, making more effective use of mission resources.

Introduction

Robots exploring other planetary bodies collect evidence to support hypotheses that have been developed from precursor (remote) sensing. It is unlikely that scientists have produced only one hypothesis explaining the preliminary data. Consequently robots exploring alien worlds are tools to determine which of a number of hypotheses best explain data observed on the surface. In this paper we present a novel algorithm for autonomous planning of scientific observations that is explicitly aware of the hypotheses the robot is investigating.

The key to scientific exploration is the falsification of hypotheses – seeking out where hypotheses disagree and experimenting at those points. Where hypotheses’ predictions agree they are equally correct or equally wrong. Sampling at points of agreement help determine the accuracy of the hypotheses. Sampling where the hypotheses disagree reveals hypothesis accuracy at the same time as determining which of the competing hypotheses are more accurate with respect to the observable universe.

Fortunately we do not necessarily need to understand semantically the inner workings of the hypotheses to choose between them, provided we can sample in their input space and make observations of their output space. In the case of geologic exploration remote scientists construct maps labelled with the expected terrain classification and robots would collect spectral observations on the ground which would then be classified and compared to the remote predictions. In this paper we represent hypotheses as the implied two-dimensional maps, but any mathematical map from an input space to an output space fits into our framework.

To illustrate the utility of our approach we simulate a planetary mission of a robot that is determining which of a number of geological classifications most accurately predict data observed on the ground. Figure 1 is a geologic map of Jupiter’s moon Io, an example of the kinds of classifications that can be produced as the result of hypothesis generated from remote sensing data. Precursor data can admit multiple explanations, implying different hypotheses about the surface. Robot observations will help resolve which of these hypotheses are correct.

In our hypothetical mission our robot is capable of moving to arbitrary locations on a map. Such a vehicle would be appropriate for a mission like exploring the Aitken basin on the south pole of Earth’s moon. The South Pole-Aitken
Basin is 2500km in diameter and as such sampling the structure requires traversing long distances. A lander vehicle capable of hopping to different locations would be ideal for this mission, and it does not have the same path planning constraints that rovers have. Our algorithm does not engage in sophisticated trajectory planning, but it could use a planner to produce informative paths for other vehicle types.

Our proposed algorithm directs robots to collect observations that quickly determine which of a set of candidate hypotheses are most likely to explain data collected in situ. The algorithm uses an information theoretic cost function that trades off between increasing the certainty in one or more of the candidate hypotheses and ensuring samples are distributed appropriately across the map.

The algorithm differs from previous science autonomy algorithms in that instead of attempting to build a model of collected data this algorithm trades off between multiple hypotheses and can determine which one is the least false. Our algorithm does not attempt to produce its own hypothesis from the observations it collects but nothing stops it from doing so. The novel contribution of this paper is the idea of hypothesis aware science autonomy. The algorithm permits robots to plan with respect to the hypotheses that are under investigation by the overall mission. Previous approaches which look at collecting representative datasets do not explicitly prioritize disagreements between competing hypotheses. The approach described in this paper does precisely that and thereby makes more effective use of resources from the outset.

Background

Science autonomy robots have not previously needed to choose between hypotheses, they have generally been focused on collecting data without prior knowledge. However, the design of experiments literature has been largely focused on selecting experiments that inform hypotheses. A popular example of an experiment selection technique is the multi-armed bandit. Likewise, model selection is a well studied field, but will not be reviewed here.

Multi-armed bandits (MABs) (Robbins 1952) are a formalization for sequentially selecting the most rewarding of a set of experiments. There are a number of different algorithms for approaching the MAB problem. The main families of which appear to Upper Confidence Bound algorithms (Lai and Robbins 1985), The Gittins Index (Gittins, Glazebrook, and Weber 2011), and Thompson Sampling (Thompson 1933). Thompson sampling has recently gained attention for being simple to implement and for having near-optimal regret properties (Chapelle and Li 2011; Agrawal and Goyal 2012; Ortega and Braun 2010). The algorithm in (Thompson et al. 2015) is aware of observations made on the ground. Its robot attempts to explain spectra data from satellites by collecting a library of spectra from the ground. Each pixel of satellite data is explained as a mixture of the endmember spectra collected by the rover. The endmembers represent a basis of examples of “pure” minerals. The rover travels to locations where library of rover-observed spectra poorly explains the satellite data and collects more observations to explain the satellite data. This approach is implicitly constructing a hypothesis about the terrain composition but the algorithm itself does not test points in the satellite data that it considers well explained, nor does it consider alternate hypotheses.

(Miller et al. 2016) uses the expected value of the Fisher Information to determine points of interest. Like mutual in-
formation (Lindley 1956), Fisher information is used as a score to select the most informative experiments. Their path planner produces smooth paths that maximize the number of high information value observations. Fisher information and the mutual information are intimately related, but not identical, quantities. However, a rigorous comparison of the behaviour of robots maximizing mutual information and those maximizing expected Fisher information does not exist in the literature. Without such evidence it is difficult to select one reward function over the other and this should be studied further.

Distributing points throughout the input space is a principled approach to learning a function de novo. However, when trying to select between competing hypotheses it is conceivable that a sequence of observation points \( x_1, \ldots, x_t \) will miss the points of conflict between competing hypotheses, especially with limited sampling resources. However, widely sampling an input space is important to ensure the predictions of a hypothesis are generally accurate. Most deployed science autonomy algorithms do not operate in the realm of hypotheses, they either focus on distributions of data or on hand-coded proxy measures, not necessarily the support or falsification of hypotheses under consideration. For that reason we propose the method described below.

**Method**

The robot explorer attempts to determine which hypothesis is most accurate, \( H^* \), taking observations from real environment \( M^* \). In order to do this the robots must select locations, \( l \in L \), in the input space of the hypotheses that most productively inform the investigation. Each hypothesis, \( H_i \), implies some mapping, \( M_i : S_{\text{in}} \rightarrow S_{\text{out}} \), from an input space, \( S_{\text{in}} \), to an output space, \( S_{\text{out}} \). In this case \( S_{\text{in}} = L \) and \( S_{\text{out}} = \{0, \ldots, K\} \), where \( K \) is the number of material classes.

The robots collect true observations, \( M^*(l) \), from the true map, \( M^* \), that can be used to update the belief in the different hypotheses based on the prediction \( M_i(l) \) from the corresponding hypothesis \( H_i \).

**Belief in Hypotheses**

The algorithms estimate their belief that hypothesis \( H_i \), where \( i = 1, \ldots, K \), is the least false with a multinomial distribution, \( P(H_i) \), where \( \sum_{i=0}^K P(H_i) = 1 \). We have a special place-holder hypothesis \( H_0 \), which represents the probability that none of the proposed hypotheses are correct.

We place a Dirichlet prior on the distribution of belief in the hypotheses, with corresponding condensation parameters \( \alpha_0, \ldots, \alpha_K \). The condensation parameters are initialized with an uninformative prior, \( \alpha_0 = 1 \), as initially all hypotheses are considered equally likely. However, the condensation parameters can be initialized to reflect any prior belief in the competing hypotheses.

After an observation the condensation parameters are updated to reflect the agreement with the predictions from the different hypotheses. That is to say:

\[
\alpha_{i,t+1} = \alpha_{i,t} + P(H_i | z_t)
\]

Which we can expand \( P(H_i | z_t) \) as follows:

\[
P(H_i | z_t) = \int_{-\infty}^{\infty} P(M_i(l_t) = x | S_t = x, z_t) \times P(S_t = x | z_t) dx
\]

Where \( S_t \) is the sensor reading taken by the robot at location \( l_t \) and \( P(S_t) \) is the sensor noise model. If we assume that the sensor noise is independent of the map implied by the hypothesis and the location the samples were collected we can continue:

\[
P(H_i | z_t) = \int_{-\infty}^{\infty} P(M_i(l_t) = x | z_t) P(S_t = x | z_t) dx
\]

If we consider that hypotheses provided by the planetary scientist do not encode uncertainty, which is likely given the authors’ interaction with planetary scientists, we can simplify this expression further:

\[
\alpha_{i,t+1} = \alpha_{i,t} + \int_{-\infty}^{\infty} \delta(x - M_i(l_t)) P(S = x | z_t) dx
\]

\[
= \alpha_{i,t} + \mathbb{P}(M_i(l_t) = z_t)
\]

Assuming that there is no uncertainty or noise in a classification is a poor assumption, if not blatantly false. However, accounting for uncertainty can be computationally expensive, which may not be compatible with space-rated hardware. Further, in this paper we are simply concerned with illustrating the utility of incorporating prior hypotheses into the sampling process, so sensor noise is eliminated to directly address the question.

In addition to the proposed hypotheses we also need to update the belief in the null hypothesis, \( H_0 \), that none of the proposed hypotheses are correct. With \( P(H_0 | z_t) \) as the probability that none of the proposed maps are correct, and assuming all \( P(H_i | z_t) \) are independent we can write as follows:

\[
P(H_0 | z_t) = P(\neg(\cup_{i=1}^K H_i) | z_t)
\]

\[
= P(\cap_{i=1}^K \neg H_i | z_t)
\]

\[
= \prod_{i=1}^K P(\neg H_i | z_t)
\]

\[
= \prod_{i=1}^K (1 - P(H_i | z_t))
\]
We observe that in the case that there is no uncertainty in either the sensor or the hypothesis then if any one hypothesis is consistent with an observation from the robot then \( P(H_0|z_i) = 0 \), and if no hypothesis is consistent with the sensor observations then \( P(H_0|z_i) = 1 \).

In this paper we consider no uncertainty in either the hypothesis or the in situ observations, so the update can be rewritten as \( \alpha_{i,t+1} = \alpha_{i,t} + \mathbb{I}(M^*(l_t) = M_i(l_t)) \), but the algorithm can incorporate map and sensor uncertainty easily.

Site Selection Algorithm

Algorithm 1 selects sampling sites with a modified version of Thompson sampling. At each time step, \( t \), the algorithm samples a belief state \( P(H_i) \forall i \in 0, \ldots, N \) from the Dirichlet prior described above, and then chooses the \( H_i \) that has the largest belief \( P(H_i) \). The algorithm assumes that \( H_i \) is “true” for the duration of this step, and uses it to evaluate which location \( l_t \) is most informative. At the location \( l_t \) the robots collect the observation \( M^*(l_t) \). Observations are used to update the belief in the hypotheses that have been assigned to them.

We assume the robot starts in the \((0, 0)\) position on the map, which corresponds to the top left-hand location of the maps shown in Figure 2. The robot is capable of sampling at locations \( l \) in the input space \( L \), which in this experiment are \((x, y)\) locations on a two-dimensional map with integer coordinates. A robot at location \( l_t \) that chooses to travel to \( l_{t+1} \) incurs a cost of \( \text{cost}(l_t, l_{t+1}) \). We assume that \( \text{cost}(l, l) = 0 \) and \( \text{cost}(l_1, l_2) > 0 \) when \( l_1 \neq l_2 \).

Once a location has been selected, the robot travels to the specified location and collects an observation from the actual map, \( M^*(l_t) \), and the belief state is updated. Should none of the hypotheses agree with the prediction of the map at \( l_t \), the place-holder condensation parameter \( \alpha_0 \) is incremented, increasing the belief that all hypotheses are equally false.

The \textit{reward} function for candidate sampling locations is how the competing algorithms control the vehicle behaviour. The reward functions for the control algorithm and our proposed algorithm are specified in Algorithm 2 and 3.

Control Algorithm - Spatial Sampling

For the control algorithm we build a density estimator using a Gaussian kernel function and we select the candidate points which the increase in information would be the greatest. We compute the entropy, \( \mathbb{H}(\cdot) \), of the kernel density estimator as described in (Beirlant et al. 1997). The algorithm for the reward function is given in Algorithm 2.

Algorithm 1 Site Selection Algorithm - The algorithm that greedily selects new sampling locations based on previous observations. The algorithm samples from the environment, \( M^* \), with the candidate hypotheses’ maps, \( M_1, \ldots, M_N \), and the total sampling budget, \( C \).

function \text{SAMPLE-SELECTION}(M^*,(M_1, \ldots, M_N),C) 
\qquad \alpha_t \leftarrow 1 \forall i \in 0, \ldots, N 
\qquad t \leftarrow 0 
\qquad l_0 \leftarrow \langle(0,0)\rangle 
\qquad \text{repeat} 
\qquad \qquad \quad P(H_i) \sim \text{Dirichlet}(\alpha_0, \alpha_1, \ldots, \alpha_K) 
\qquad \qquad \quad i_t \leftarrow \arg\max_{i \in \{0, \ldots, N\}} P(H_i) 
\qquad \qquad \quad l_t \leftarrow \arg\max_{l \in L} \text{reward}(P(H_i), M_i, \langle l_0, \ldots, l_{t-1} \rangle) 
\qquad \qquad \quad \text{for} i \leftarrow 1, K \text{ do} 
\qquad \qquad \qquad \alpha_i \leftarrow \alpha_i + \mathbb{I}(M^*(l_t) = M_i(l_t)) 
\qquad \qquad \text{end for} 
\qquad \text{if} \sum_{i} \mathbb{I}(M^*(l_t) = M_i(l_t)) = 0 \text{ then} 
\qquad \quad \alpha_0 \leftarrow 1 
\qquad \text{end if} 
\qquad C \leftarrow C - \text{cost}(l_t) 
\qquad t \leftarrow t + 1 
\qquad \text{until} \quad \text{budget} = 0 
\text{end function}

This algorithm takes as an argument the belief state and the map corresponding to the hypothesis with the maximum belief state, but they are not used in calculating the reward value. These arguments are provided to the function to ensure type consistency with Algorithm 3.

Proposed Algorithm - Hypothesis Falsification

This algorithm seeks sampling locations that concentrate belief across the hypotheses given a predicted observation \( M_i(l_t) \), as seen in equation 13, where \( M_i \) is the map associated with the “true” hypothesis as per Algorithm 1. These locations maximize the mutual information between the belief state \( P(H_i) \) and the predicted observation \( M_i(l_t) \).

\[
I(P(H_i); z_t = M_i(l_t)) = \mathbb{H}[P(H_i)] - \mathbb{H}[P(H_i|z_t = M_i(l_t))] \tag{13}
\]

This will automatically seek out locations where the competing \( H_i \)’s disagree. The objective function that is being maximized focuses on collecting observations that concentrate belief in one of the hypotheses, that is to say to reduce the entropy post-observation. Since cases where the hypotheses agree in their predictions will share belief, the points where the “true” hypothesis disagrees with the competing algorithms stands the best chance of concentrating the belief. With equation 13 as the reward function at any point \( l \) where the \( M_i \)’s agree will increase the credibility of all hypotheses, resulting in negative mutual information for \( l_t \).

However, one also wants to ensure that the hypotheses are accurate. Sampling at disagreement points will help build
the credibility of the best hypothesis, but it does not give confidence with the overall predictions of \( H_i \). To mitigate this problem we add to equation 13 the result computed in Algorithm 2 to ensure the sampled points spatially diverse. The reward function used for this algorithm is given in Algorithm 3.

**Algorithm 3** Hypothesis Falsification Sampling. We use equation 13 to estimate the informativeness, with respect to hypothesis selection, of different points in the map. That informativeness is added to the value of points for their spatial diversity.

```plaintext
function HFS-reward(\( P(H_i), M_i, \langle l_0, \ldots, l_{t-1}, l_t \rangle \))
    \( P(H_i|z_t = M_i(l_t)) \leftarrow \text{update}(P(H_i), M_i(l_t)) \)
    \( r_h \leftarrow \mathbb{H}[P(H_i)] - \mathbb{H}[P(H_i|z_t = M_i(l_t))] \)
    \( r_s \leftarrow \text{spatial-reward}(P(H_i), M_i, \langle l_0, \ldots, l_{t-1}, l_t \rangle) \)
    return \( r_h + r_s \)
end function
```

**Map Generation**

We generated 10 different maps, each \( 20 \times 20 \) pixels, with each pixel containing a label from \( z \in 1, \ldots, K \). We produce the ground truth maps by selecting 20 seed locations and randomly assigning them a label \( z \) with uniform probability (denoted by the function \( U(z) \)), over the \( K \) labels. We then use a Voronoi map generation algorithm, given in Algorithm 4. Example maps are shown in Figure 2a.

**Algorithm 4** Map Generation Algorithm - Produces maps that are labelled grids of \( nPixels \times nPixels \) cells. The map uses a Voroni generation algorithm from \( nSeeds \) seed points assigned one of \( nLabels \) randomly generated labels.

```plaintext
function GENERATE-MAP(nSeeds,nPixels,nLabels)
    seeds ← \( \emptyset \)
    map ← zeros(nPixels,nPixels)
    for \( i \in 1, \ldots, nSeeds \) do
        \( x_i \sim U(nPixels) \)
        \( y_i \sim U(nPixels) \)
        label ← \( U(nLabels) \)
        seeds ← seeds + (\( (x_i, y_i, label) \)).
    end for
    for \( x \in 1, \ldots, nPixels \) do
        for \( y \in 1, \ldots, nPixels \) do
            map\( (x,y) \leftarrow \text{closest}(seeds, x, y, label) \)
        end for
    end for
    return map
end function
```

The hypotheses the robots were attempting to validate were generated by corrupting generated map data. This is a stand-in for hypotheses that have different interpretations of the precursor data. To generate the hypotheses maps we take the same 20 seed points used to generate a true map and mis-label the seed locations with probability \( P(z = i|z = j) = \epsilon/(K-1) \) for all \( i \neq j, i, j \in 1, \ldots, K \). Using the corrupted seeds we then generate maps using the Algorithm 4. Figures 2b, 2c, 2d shows the effect of maps as \( \epsilon \) increases. We tested hypotheses generated with \( \epsilon \in \{0.1, 0.2, 0.3, 0.5, 0.7, 0.8, 0.9\} \) which correspond to \( H_1 \) to \( H_7 \) in Table 1. For simplicity of experiments we set \( K = 2 \) and assumed no sensor noise on the part of the robot, but again the algorithm admits more complex scenarios.

**Experiments**

We compared the performance of our proposed algorithm and of the control algorithm in three experiments. For each experiment we tested ten different maps, and each map was repeated twenty-four times. The true maps and the hypothesis maps were generated as described above. The similarity between the true map and the hypotheses are given in Table 1. The similarity measure was the prediction accuracy of the hypothesis map:

\[
\text{similarity}(i, j) = \frac{\sum_{l \in L} 1(M_i^*(l) = M_j(l))}{|L|} \tag{14}
\]

When a robot travelled to a point on the map it was informed of the true label of the point without error.

We assume in this experiment that the robot is capable of hopping long distances. As such the cost of traversal is simply limited to the number of hops. The robots were given a maximum budget of 100 hops. Since the most hazardous parts of a hop are the take-off and landing, we make the assumption that the associated risk of crashing dominates the traverse cost. Consequently, the cost function is the constant function \( cost(\cdot, \cdot) = 1 \). However, missions with different mobility strategies would incur different costs and the cost function can be modified to reflect that.

In Experiment 1 the hypotheses are all of reasonable quality. This represents a mission with good precursor data and generated hypotheses. In Experiment 2 the hypotheses are of mixed quality. In this scenario the precursor data is ambiguous, but there is one good hypothesis. In Experiment 3 all hypotheses are of poor quality. In this scenario the hypotheses are not fitted to the environment.
The True map for $M_0$. 

(b) $H_1$, 93.25% similarity with $M_0$, $\epsilon = 0.1$. 

(c) $H_4$, 49% similarity with $M_0$, $\epsilon = 0.5$. 

(d) The $H_{10}$, 7.25% similarity with $M_0$, $\epsilon = 0.9$. 

Figure 2: Examples of a map and the hypotheses generated that anticipate the map.

Table 2: The hypotheses tested in the experiments. Column $H^\dagger$ gives the best hypothesis for each experiment. Subscripts correspond to Table 1.

<table>
<thead>
<tr>
<th>Experiment</th>
<th>$H^\dagger$</th>
<th>Distractors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$H_1$</td>
<td>$H_2$ $H_3$</td>
</tr>
<tr>
<td>2</td>
<td>$H_1$</td>
<td>$H_4$ $H_7$</td>
</tr>
<tr>
<td>3</td>
<td>$H_5$</td>
<td>$H_6$ $H_7$</td>
</tr>
</tbody>
</table>

Analysis of Results

At the end of the experiments we compute mean of $P(H^\dagger)$. Across the ten different maps we computed the effect size of our intervention. To determine the effect size we use Cohen’s $d$, with pooled standard deviation, $\sigma_{pooled}$.

\[
d = \frac{\mu_1 - \mu_0}{\sigma_{pooled}}
\]

where $\mu$ and $\sigma$ are the sample mean and standard deviation of $P(H^\dagger)$ for the control (0) and proposed (1) algorithms, respectively. Cohen’s $d$ has four levels for effect size: $d < 0.2$ is a negligible effect, $0.2 < d < 0.5$ is a small effect, $0.5 < d < 0.8$ is a medium effect, and $d > 0.8$ is a large effect. Negative values of $d$ mean that the control algorithm has a higher belief in $H^\dagger$. There are criticisms of using these standards (Lipsey et al. 2012), but the dearth of effect size reporting in robotics does not provide alternative thresholds for effect size magnitudes.

Results

Figure 3 shows the average belief state for the two different algorithms. In Experiment 1 our proposed algorithm was able to converge on $H^\dagger$ with high confidence. Our algorithm achieved a statistically significant ($p < 0.01$) large improvement ($d = 1.71$) over the control algorithm. We consider this experiment to be the most representative of mission scenarios, as hypotheses generated from precusor data should have some accuracy.

In Experiment 2 our algorithm outperforms the control, shown in Figure 4. Both converge on $H_1$ being the most credible, followed by $H_4$ and $H_7$. The control algorithm gives marginally more weight to the belief that none of the hypotheses are correct, $H_0$. Our algorithm outperforms the control algorithm in this experiment with $p < 0.01$ and $d = 1.84$. This is a larger effect size than Experiment 1, due to the greater disparity in accuracy of the hypotheses. Our algorithm amplifies this disparity by testing on where the algorithms disagree.

In the third experiment all the hypotheses predict the true terrain poorly. The belief state at the end of sampling is shown in Figure 5. Again, our algorithm selects $H^\dagger$, with significant improvement over the control, $p < 0.01$, $d = 1.54$. On the other hand, Figure 5 shows that the control al-
Belief

**Control**

Control

Proposed

Proposed

Figure 4: Experiment 2 results. Our proposed algorithm again outperforms the control algorithm with statistical significance. Error bars are one standard error, \( N = 10 \).

algorithm is significantly more confident that none of the algorithms are correct. None of the hypotheses in this experiment have a prediction accuracy greater than 50\%, \( H_0 \) - no correct hypotheses - is a valid conclusion. Our algorithm had more belief in \( H_0 \) than previously, but the control algorithm had statistically significantly higher \( P(H_0) \), \( p < 0.01 \), \( d = 5.55 \). We consider this scenario to be unlikely in planetary missions, but it does expose a weakness in our algorithm that should be addressed for when a lack of precursor data could lead to poor hypotheses.

Conclusions

We achieved our goal of developing an algorithm that can identify the least false of a set of competing hypotheses. Our algorithm is a significant improvement over a standard exploration algorithm. When the candidate hypotheses were of poor quality our algorithm was still able to pick the best one. However, the control algorithm better identified that none of the hypotheses were good. As the obvious extension of this work is to simultaneously generate and falsify hypotheses - one hypotheses must always be correct. This problem may vanish when hypotheses’ predictions are high dimensional, like spectrometer readings or neural network outputs.

Sensor noise and hypothesis uncertainty were not addressed in this work, but are an important part of decision making. They will be incorporated in future experiments. By liberating our algorithm from path planning we demonstrated the utility of hypothesis-aware value functions for automate exploration. This work will be integrated into a planner for wheeled vehicles in future experiments. So equipped our exploring robots will be able to better serve us as they travel farther into space.

Acknowledgements This research was funded by the ARADS project under the P-STAR program.

References


