Sub-linear Algorithms for Landmark Discovery from Black Box Models

Manan Sanghi[*] and **Praveen Paritosh**[†] and **Reuben Thomas**[‡]

Northwestern University, Evanston, IL 60201, USA.
{manan,paritosh,r-thomas}@northwestern.edu

Abstract

Most models of qualitative reasoning depend upon qualitative representations of quantity that make the necessary and relevant distinctions for the reasoning task at hand. Automatically generating such abstractions from numerical models has been pointed out to be a practically significant and potentially difficult problem [Struss, 2003]. Previous work [Sachenbacher and Struss, 2001] used finite relational models as a starting point to generate abstractions. In this paper, we work with a black box model that relates an output variable with known landmarks to a set of input variables for which the landmarks need to be determined. For most problems of practical significance, the input space is too large to be exhaustively examined. We present a simple randomized scheme for discovering landmarks which performs surprisingly well in time that is only polylogarithmic in the input size.

1 Introduction

A key insight of qualitative reasoning is that powerful reasoning can be performed with an appropriate quantization of the continuous space. In the *quantity space* representation [Forbus, 1984], continuous values are represented via sets of ordinal relationships to interesting comparison points. There are two kinds of such comparison points. *Limit points* are derived from general properties of a domain as applicable to a specific situation. The precise numerical values of these limit points can change over time, e.g., the boiling point of a fluid is the function of its pressure. *Landmark values* denote constant points of comparison on the space of numerical values.

By letting the modeler choose these comparison points, the quantity space representation allows for variable resolution, to make just the necessary and relevant distinctions for the reasoning task at hand. For example, the temperature of a fluid might be represented in terms of its relationship to the freezing and boiling points of the fluid. The particular comparison points are usually chosen by the modeler as a first step to writing qualitative model fragments. The problem of how to automatically find the necessary and relevant distinctions remains largely unsolved (but see [Sachenbacher and Struss, 2001] [Paritosh, 2003]).

[Struss, 2003] has pointed out the practical importance and difficulties in generating such abstractions automatically from numerical simulation models. In a real-life industrial scenario, one might have access to complex and opaque numerical models – MATLAB/Simulink models with nonlinear analytic functions, tables with empirical data and even black-box model fragments with C code. Transforming such a model into a qualitative diagnosis model provides finite compact representations that can be used for on-board diagnosis.

In this paper, we present the Landmark Discovery (LD) problem, and randomized algorithms which solve it with provable performance guarantees. The time complexity of our algorithms is only polylogarithmic in the input size with polynomially small error probability.

We motivate the problem with an example in Section 2. Section 3 is devoted to definitions and terminology. Section 4 presents the problem formulation, algorithm and analyses. Section 5 discusses related work. We conclude with future work in section 6.

2 Black Box Landmark Discovery

Let's look at a simple example. Consider the case of fluid flow through a pipe. At low velocities, the flow is smooth, or laminar. Depending on the ratio of inertial and viscous forces, which is captured by Reynold's number, the flow can be laminar, transitional, or turbulent.

Suppose for a certain flow, we are given a black-box model, M, that relates the Reynold's number, R, in a certain flow to the velocity of flow, V, the characteristic distance describing the flow, D, viscosity of fluid, μ, and the density, ρ. This is used for the sake of illustration, as for certain flows one might have a closed-form expression for Reynold's number.

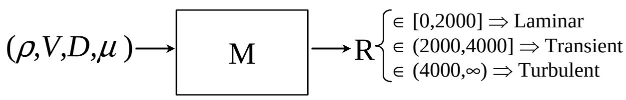


Figure 1: Black box model for computing Reynold's number

[*]Department of Computer Science
[†]Department of Computer Science
[‡]Industrial Engineering and Management Science

In this model, R is the output variable, and we can query the model with values for all the input variables, namely, V, D, μ and ρ. For each of the input variables, we are given the range of values that they can take, and the granularity. The interesting distinctions for the values of Reynold's number characterizing the flow are given to us as $\{(0, 2000), (2000, 4000), (4000, \infty)\}$ with the three intervals corresponding to laminar, transitional and turbulent flow scenarios. Following [Sachenbacher and Struss, 2001], we call these the *target distinctions*. We are interested in finding the corresponding distinctions for the input variable. The range and the granularity of input variables gives rise to a discrete input space. This space can be very large. Not all distinctions in the input space are needed if we are just interested in type of flow. Given a black box model and a set of target qualitative distinctions for the output variable, we are interested in finding the coarsest representation of the input space, i.e. the minimum distinctions that we need to make in order to capture all the distinctions that M makes.

One such representation is a set of *landmarks* for each of the input variables. If there are d input variables, the landmarks imply a grid whose cells are d-dimensional hyper-rectangles such that for any point inside this hyper-rectangle, the output variable is in the same target qualitative state.

3 Definitions and Terminology

We consider a system with one output variable, y, and d input variables. The discussion here can be generalized to the case of more than one output variables. We assume that there is a model, M, which has a functional form, i.e., $y = M(x_1, x_2, \ldots, x_d)$. We say that M is a black box model as we don't know M directly, or make any assumptions about M. M could be instantiated as Simulink/C code. We can *query* M with values for the input variables to find the value for the output variable.

We assume that input variable, x_i can take real values from a given closed interval. Even though input variables can take real values, because of measurement and/or observability limitations, we have a maximum granularity on the input values. A measurement granularity is the smallest difference that can be noticed. Thus the domain of input variables is observable as a set of discrete points in the given interval. In Sachenbacher and Struss' formalism, this corresponds to the set of observable distinctions for the variable. Let the domain of an input variable, x_i, be the set $\mathcal{I} = \{1, 2, \ldots, n\}$. We assume the cardinality of each of the dimensions to be the same for the ease of exposition. However, this assumption is not critical for either the correctness or performance of our algorithm. The output variable takes on real values. Furthermore, we are given a partition of the domain of output variable, which correspond to qualitatively distinct regions called the *target distinctions*.

By querying the model, M, with values for the input variables we can obtain the value for the output variable, and thus the corresponding target distinction. Thus M implies a mapping, f, from the discrete input space to the discrete output space of target distinctions. Let τ be the set of given *target distinctions* for the output variable. Note that τ is countable and finite. *Landmarks* are points in the domains of each input variable. The output variable belongs to two different target distinctions across a landmark of a given variable, for some combination of input values of the other variables. We represent the landmarks for the input variable x_j as the landmark set, $\mathcal{L}^j = \{\ell_1^j, \ldots, \ell_m^j\}$. A landmark set is called *maximal* if it contains all the landmarks for that input variable. We illustrate these concepts in Figure 2(a) for the case of one input and one output variable.

The *landmark discovery* problem is to find the maximal landmark sets for each of the input variables. In the next section we formally define this problem.

4 Algorithms and analysis

We first present the simpler case of one input variable to illustrate the algorithm, after which we discuss the general case of d input variables.

4.1 Landmark Discovery with one input variable

Problem 1 (Landmark Discovery: 1 Input).
INPUT*: A function* $f : \mathcal{I} \to \tau$.
OUTPUT*: A set of points* $\mathcal{L} = \{\ell_1, \ldots, \ell_m\}$ *such that* $f(\ell_i - 1) \neq f(\ell_i) \forall i$ *and* $|\mathcal{L}|$ *is maximized.*

Let n be the number of points in the input space, i.e., $n = |\mathcal{I}|$. Let $\mathcal{L}_* = \{\ell_1, \ldots, \ell_m\}$ be the true landmark set such that $\ell_1 < \ell_2 < \cdots \ell_m$. We now present a randomized algorithm which outputs a landmark set $\mathcal{L}_{\text{out}} \subseteq \mathcal{L}_*$ such that $\mathcal{L}_{\text{out}}$ contains all the landmarks in $\mathcal{L}_*$ with high probability.

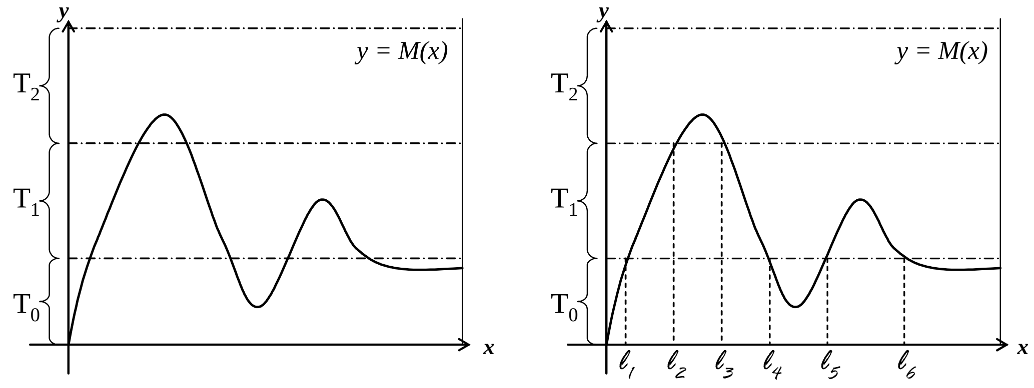
Algorithm 1 *1-LD*(c,δ)

1. Sample f uniformly at $s = \frac{c \cdot \log n}{\delta}$ points from $\mathcal{I}$. Let the points be $r_1, r_2, \ldots, r_s$ such that $r_1 \leq r_2 \leq \ldots \leq r_s$.
2. Let $\mathcal{L}_{\text{out}} = \emptyset$.
3. For all $i \in [1, s-1]$, if $f(r_i) \neq f(r_i + 1)$, do a binary search to find a landmark ℓ such that $f(\ell) \neq f(\ell - 1)$. Let $\mathcal{L}_{\text{out}} = \mathcal{L}_{\text{out}} \cup \{\ell\}$.
4. Output $\mathcal{L}_{\text{out}}$.

Theorem 1. *Algorithm 1-LD finds all landmarks which are at least* $\delta{\cdot}n$ *apart in* $O(\frac{c}{\delta} \log^2 n)$ *runtime with error probability* $o(m/n^c)$ *for any constant* $c > 0$.

PROOF: For any two consecutive sample points, Algorithm 1-LD spends at most $O(\log n)$ time for binary search. Since there are a total of $O(\frac{c}{\delta} \log n)$ sample points, the total runtime of algorithm is $O(\frac{c}{\delta} \log^2 n)$.

For any $\ell_j \in \mathcal{L}_*$ such that $\ell_j - \ell_{j-1}, \ell_{j+1} - \ell_j > \delta{\cdot}n$, let P_{ℓ_j} be the probability of not including ℓ_j in $\mathcal{L}_{\text{out}}$. Note that if the set of sample points contain some $x_\alpha \in [\ell_{j-1}, \ell_j]$ and $x_\beta \in [\ell_j, \ell_{j+1}]$, then we are guaranteed to include ℓ_j in $\mathcal{L}_{\text{out}}$.



(a) (b)

Figure 2: Example of the landmark discovery problem for one input variable. Figure 2(a) shows the relationship between the input variable x and the output variable y. T_0, T_1, T_2 are target distinctions for y. Figure 2(b) shows the intervals on the domain of x identified by the *landmarks*, $\ell_1, \ell_2, \ldots, \ell_6$ corresponding to the given target distinctions. All points within an interval between two adjacent landmarks on x map to the same target distinction on y. Across a landmark, the variable x maps to different target distinctions.

Therefore,

$$\begin{aligned} P_{\ell_j} &\leq \Pr[\nexists \text{ a sample in } [\ell_{j-1}, \ell_j] \\ &\qquad \text{or } \nexists \text{ a sample in } [l_j, l_{j+1}]] \\ &\leq (1 - \frac{\ell_j - \ell_{j-1}}{n})^s + (1 - \frac{\ell_{j+1} - \ell_j}{n})^s \\ &\leq 2\cdot(1 - \frac{\delta n}{n})^s \\ &= 2\cdot(1-\delta)^{\frac{c\cdot \log n}{\delta}} \\ &= 2\cdot n^{\frac{c \log(1-\delta)}{\delta}} \\ &< 2\cdot n^{-c}. \end{aligned}$$

Since $P_{\ell_j} < 2\cdot n^{-c}$ for all $\ell_j \in \mathcal{L}_*$, the probability that $\mathcal{L}_{\text{out}}$ misses any of the landmarks in $\mathcal{L}_*$ which are at least $\delta\cdot n$ apart is at most $2\cdot m\cdot n^{-c}$ which is $o(m/n^c)$. ■

Corollary 1.1. *If all landmarks in $\mathcal{L}_*$ are at least $\delta\cdot n$ apart, then Algorithm 1-LD finds them all in time $O(\frac{c}{\delta}\log^2 n)$ with error probability $o(m/n^c)$ for any constant $c > 0$.*

4.2 Landmark Discovery with d input variables

Problem 2. *[Landmark Discovery: d Inputs]*
INPUT: *A function $f : \mathcal{I}^d \rightarrow \mathcal{A}$.*
OUTPUT: *Sets $\mathcal{L}^1, \ldots, \mathcal{L}^d$ where $\mathcal{L}^j = \{\ell^j_1, \ldots, \ell^j_m\}$ such that the following holds for all $1 \leq j \leq d$*

1. *For all $\ell^j \in \mathcal{L}^j$, $\exists x^1, \ldots, x^{j-1}, x^{j+1}, \ldots, x^d \in \mathcal{I}$ such that $f(x^1, \ldots, x^{j-1}, \ell^j - 1, x^{j+1}, \ldots, x^d) \neq f(x^1, \ldots, x^{j-1}, \ell^j, x^{j+1}, \ldots, x^d)$*
2. *$|\mathcal{L}^j|$ is maximized.*

Let $n = |\mathcal{I}|$. The total size of the input space is n^d. Let $\mathcal{L}^j_* = \{\ell^j_1, \ldots, \ell^j_m\}$ be the true landmark set. We present a randomized algorithm which outputs a landmark set $\mathcal{L}^j_{\text{out}} \subseteq \mathcal{L}^j_*$ such that $\mathcal{L}^j_{\text{out}}$ contains all the landmarks in $\mathcal{L}^j_*$ with high probability.

For the d-dimensional case, the landmarks imply a grid whose cells are d-dimensional hyper-rectangles such that for any point inside this hyper-rectangle, the output variable is in the same target qualitative state.

Let $\vec{a} = (a^1, \ldots, a^d)$ denote a point in d-dimensional space and $\vec{a}^j$ be its j^{th} component a^j. Each landmark $\ell^j_i \in \mathcal{L}^j_*$ defines a $d-1$ dimensional axis parallel hyperplane $H_{\ell^j_i}$ given by the equation $\vec{x}^j = \ell^j_i$. Further let $A_{\ell^j_i}$ and $B_{\ell^j_i}$ be two adjacent grid cells such that their common face lie on $H_{\ell^j_i}$ and the points in $A_{\ell^j_i}$ belong to a different target qualitative state than those in $B_{\ell^j_i}$, i.e. if $\vec{a} \in A_{\ell^j_i}$ and $\vec{b} \in B_{\ell^j_i}$, then $f(\vec{a}) \neq f(\vec{b})$. We call any such $A_{\ell^j_i}$ and $B_{\ell^j_i}$ to be ℓ^j_i-*separated grid cells*.

Definition For any two points $\vec{x}$ and $\vec{y}$ such that $f(\vec{x}) \neq f(\vec{y})$, a landmark ℓ^j is said to *resolve* $\vec{x}$ and $\vec{y}$ if and only if $\vec{x}^j \leq \ell^j < \vec{y}^j$ or $\vec{x}^j < \ell^j \leq \vec{y}^j$.

Figure3 illustrates these definitions for the case of two input variables.

Algorithm 2 *d-LD*(c,Δ)

---

1. Sample f uniformly at $s = \frac{c}{\Delta}\log n^d$ points from $\mathcal{I}^d$.
2. Let $\mathcal{L}^j_{\text{out}} = \emptyset \quad \forall\, j$.
3. For all pairs of sample points s_a and s_b, if $\nexists \ell^j_i \in \mathcal{L}^j$ for any j such that ℓ^j_i resolves s_a and s_b, then do a binary search between s_a and s_b to find a landmark $\ell^{j'}_{i'}$ which resolves them. Let $\mathcal{L}^{j'}_{\text{out}} = \mathcal{L}^{j'}_{\text{out}} \cup \{\ell^{j'}_{i'}\}$.
4. Output $\mathcal{L}^j_{\text{out}} \quad \forall\, j$.

Theorem 2. *Algorithm d-LD finds all landmarks ℓ such that $\ell-$separated grid cells are at least $\Delta\cdot n^d$ large in*

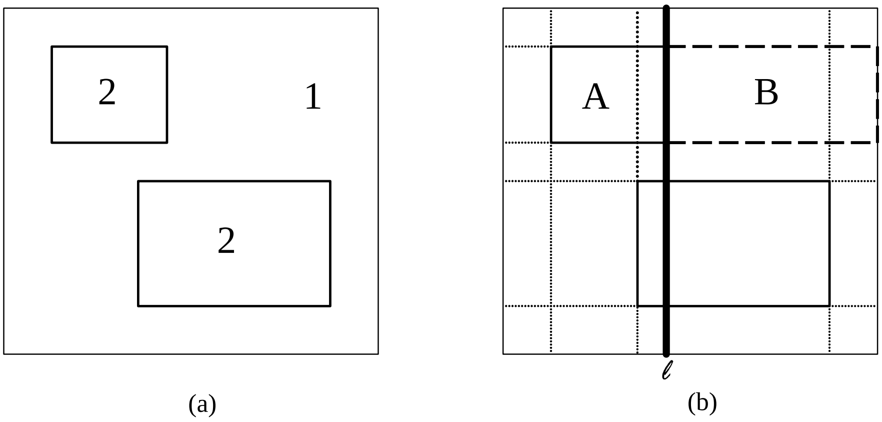


Figure 3: Example illustrating the concepts used in the d dimensional landmark discovery problem. Shown is the input space for two inputs. Figure 3(a) depicts regions which correspond to different target qualitative distinctions, 1 and 2. The dotted lines in Figure 3(b) indicate the grid implied by these regions. The landmark, ℓ defines the 1 dimensional hyperplane (line), represented by the thick line. A and B represents the ℓ-separated grid regions.

$O(m(\frac{cd}{\Delta})^2 \log^2 n)$ *runtime with error probability* $o(m/n^{cd})$ *for any constant* $c > 0$.

PROOF: For any pair of sample points, we spend at most $O(m)$ time searching for a resolving landmark. For every landmark, we spend at most $O(d \log n)$ time for the binary search in which it is discovered. Hence the total running time is $O(ms^2 + md \log n)$ which is $O(m(\frac{cd}{\Delta})^2 \log^2 n)$.

For any $\ell_i^j \in \mathcal{L}_*^j$, let A and B be two ℓ_i^j-separated grid cells. Note that a point in A and one in B can be resolved only by the landmark ℓ_i^j. Hence, we are guaranteed to include ℓ_i^j in $\mathcal{L}_{\text{out}}^j$ if the set of sample points contain some $x_\alpha \in A$ and $x_\beta \in B$. Therefore, if $|A| \geq \Delta{\cdot}n^d$ and $|B| \geq \Delta{\cdot}n^d$, then

$$\begin{aligned}
P_{\ell_i^j} &\leq \Pr[\nexists \text{ a sample in } A \text{ OR } \nexists \text{ a sample in } B] \\
&\leq \left(1 - \frac{|A|}{n^d}\right)^s + \left(1 - \frac{|B|}{n^d}\right)^s \\
&\leq 2{\cdot}\left(1 - \frac{\Delta{\cdot}n^d}{n^d}\right)^s \\
&= 2{\cdot}(1-\Delta)^{\frac{c \cdot \log n^d}{\Delta}} \\
&= 2{\cdot}n^{\frac{cd \log(1-\Delta)}{\Delta}} \\
&< 2{\cdot}n^{-cd}.
\end{aligned}$$

Since $P_{\ell_i^j} < 2{\cdot}n^{-c}$ for all $\ell_i^j \in \mathcal{L}_*^j$ for all dimensions j, the probability that $\mathcal{L}_{\text{out}}$ misses any of the landmarks $\ell^j \in \mathcal{L}_*^j$ for any j such that the ℓ^j-separated grid cells are at least $\Delta{\cdot}n^d$ in size is at most $2{\cdot}m{\cdot}n^{-cd}$ which is $o(m/n^{cd})$. ■

4.3 Discussion

The input space I^d is too large to be exhaustively searched for landmarks. In an adversarial situation, scanning the entire input space is unavoidable. However, the above algorithms demonstrate that we can get good guarantees for finding all the landmarks in sublinear time when the landmarks are not too close to each other. Such an assumption is quite reasonable in practice. For instance, based on physical or measurement constraints, we might expect how close the landmarks can be. The constant δ captures this constraint. The constant c represents the tradeoff between running time and probabilistic guarantee of success.

5 Related Work

Although the idea of necessary and relevant distinctions is a cornerstone of qualitative reasoning, Struss and Sachenbacher were the first to highlight and formalize the problem as the *Qualitative Abstraction Problem* [Struss and Sachenbacher, 1999]. They gave a solution to the case of finite relational models, and an implementation of their algorithm, AQUA.

The problem presented here is a special case of the qualitative abstraction problem for the case of ordered domains. The domain $\mathcal{I}^d$ of the function f maps to the concept of *observable distinctions*. The *domain abstractions* are captured by the sets, $\mathcal{L}$. The *target distinctions* are captured by the set τ.

The *target distinctions* are present only on the single output variable (in our formulation). Our algorithm extends easily to the case when there are target distinctions on more than one output variable. We find the *domain abstractions*, $\mathcal{L}$ for each variable (with target distinctions) separately and then merge (find intersections) of the results. This statement has actually been proved in [Struss and Sachenbacher, 1999].

The problem formulation in this work prescribes a functional relationship that connects each variable with target distinctions with the other variables. The requirements that the resultant solution be *distinguishing* and *maximal* is captured by conditions 1 and 2 in Problem 2 in Section 4. While the solution methodology described in [Sachenbacher and Struss, 2001] applies to general case of unordered domains, the work here presents an efficient way of solving the problem with ordered domains. Our approach could be used in conjunction with their model-based approach which exploits knowledge of relationships between variables. Or, one could use our methods to create a finite abstraction of the input space that could be then used as a starting point by a system like AQUA. In our problem we assume that at least one of the output variables is also the target variable.

Another very different approach is taken by [Paritosh, 2003]. The goal of his work is to find cognitively plausible qualitative representations of quantity. The key insight there is that important qualitative distinctions arise because of discontinuities in the relational structure of the domain. The theory has been implemented in a system, CARVE, that takes a set of examples represented in predicate calculus as input and determines the *limit points* on various quantitative dimensions.

6 Conclusions and Future Work

Clearly, no algorithm can guarantee to find all the landmarks without looking at the entire input space. However the input space could be prohibitively large to be exhaustively examined. For such cases we are able to find landmarks that are not too close to each other with polynomially small error probability in polylogarithmic runtime.

In this paper we have only analyzed the case when the qualitative states correspond to axis-parallel hyper-rectangles in the input space. As future work, we intend to extend these techniques to general d-dimensional polyhedra. We also believe there is scope for tightening the analysis and improving the run-time of the algorithm by more carefully choosing a subset of all the pairs of sample points to be resolved.

We presented sublinear algorithms for finding all landmarks under the assumption that they are not too close. Another possible approach might be to allow the landmarks to be arbitrarily close but exploit the property that the number of landmarks for a variable is usually much less than its input space.

Acknowledgments

Praveen Paritosh would like to thank Ken Forbus and Artificial Intelligence Program of the Office of Naval Research for support. Manan Sanghi would like to thank Ming-Yang Kao. The authors would like to thank Peter Struss and the reviewers for helpful comments.

References

[Forbus, 1984] Kenneth D. Forbus. Qualitiative Process Theory. *Artificial Intelligence*, 24:85–168, 1984.

[Paritosh, 2003] Praveen K. Paritosh. A Sketch of a Theory of Quantity. In *Proceedings of the* 17^{th} *International Workshop on Qualitative Reasoning, Brasilia, Brazil*, 2003.

[Sachenbacher and Struss, 2001] Martin Sachenbacher and Peter Struss. AQUA: A Framework for Automated Qualitative Abstraction. In *Proceedings of the* 15^{th} *International Workshop on Qualitative Reasoning, San Antonio, USA*, 2001.

[Struss and Sachenbacher, 1999] Peter Struss and Martin Sachenbacher. Significant Distinctions Only: Context-Dependent Automated Qualitative Modeling. In *Proceedings of the* 13^{th} *International Workshop on Qualitative Reasoning, Loch Awe, Scotland*, 1999.

[Struss, 2003] Peter Struss. Automated Abstraction of Numerical Simulation Models - Theory and Practical Experience. In *Proceedings of the Workshop of Algebrain Models of Reasoning, KI-2003, Hamburg, Germany*, 2003.