The Computational Complexity of Inverse Integer Programming

Piotr Wojciechowski*
LCSEE,
West Virginia University,
Morgantown, WV
{pwojciec@mix.wvu.edu}

K. Subramani
LCSEE,
West Virginia University,
Morgantown, WV
{K.Subramani@mail.wvu.edu}

Abstract

The main purpose of this paper is to determine the computational complexity of the Inverse Integer Programming Problem. This differs from regular optimization in which you are given the optimization function and solution space and are asked to find an element contained within the solution space that either minimizes or maximizes the given function. In the case of Integer Programming the regular optimization problem consists of finding a point $\mathbf{x}$ in $\mathbb{Z}^n$ that satisfies a system of constraints and either minimizes or maximizes the function $f(\mathbf{x}) = \mathbf{c} \cdot \mathbf{x}$ for a given vector $\mathbf{c}$. For the inverse problem we are given the point $\mathbf{x}$, the system of constraints, and a target vector $\mathbf{c}$ and the goal is to find a vector $\mathbf{c}'$ such that $\mathbf{x}$ either minimizes or maximizes $\mathbf{c}' \cdot \mathbf{x}$ and which is as close to the target vector $\mathbf{c}$ as possible. We will show that this problem is Δ_2^P-hard. This extends the results in [] which show that this problem is contained within Δ_2^P.

1 Introduction

A normal optimization problem consists of finding the optimal input, which must meet a set restriction, to a function. In the corresponding inverse optimization problem we are given an input, which satisfies the same set of restrictions, and a target function. We are then expected to find a function for which the given input is optimal and is which is closest to the target function. The method through which functions are compared to determine closeness depends on the problem.

Having this requirement of being close to a target function is necessary to prevent a constant function being produced as an output. Since such a function has the same value for every input it would be trivially optimal for the given input.

In general Inverse Optimization is used in problems where we know the outcome but want to find out what factors contributed to it.

The principal contributions of this paper is establishing that Inverse Integer Programming is Δ_2^P-hard.

The rest of this paper is organized as follows: Section 2 provides a formal description of the problem under consideration. The motivation for our work is described in section 3. Related papers are discussed in Section 4. In Section 5 we prove our complexity results. Finally we conclude in section 6 by summarizing our contributions and identifying avenues for future research.

2 Statement of Problem

In this section we define the problem at hand. We will start by defining Integer Programming.

*This research was supported in part by the National Science Foundation through Award CCF-1305054 and Award CCF-0827397, and Air Force Office of Scientific Research through Award FA9550-12-1-0199.

Definition 2.1 *An* Integer Constraint *on the vector* $\mathbf{x}$ *consists of a vector* $\mathbf{a}$ *and integer* b*, and is said to be satisfied by* $\mathbf{x} \in \mathbb{Z}^n$ *if* $\mathbf{c} \cdot \mathbf{x} \leq b$*.*

Example (1): $2 \cdot x_1 + 3 \cdot x_2 \leq 1$ is an integer constraint where $\mathbf{a} = (2,3)$ and $b = 1$. It is satisfied by the point $\mathbf{x} = (0,0)$ but not by the point $\mathbf{x} = (1,1)$.

Definition 2.2 *A* system of integer constraints $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$*. is a collection of multiple integer constraints. The system* $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$ *is satisfied by* $\mathbf{x} \in \mathbb{Z}^n$ *if for each constraint in the system* $\mathbf{A_i} \cdot \mathbf{x} \leq b_i$*.*

Example (2): Consider the system of integer constraints

$$\begin{aligned} x_1 + 2 \cdot x_2 &\leq 3 \\ x_1 + x_2 &\leq -1 \\ -x_1 &\leq 0 \end{aligned}$$

The point $\mathbf{x} = (1,-2)$ satisfies each constraint and thus satisfies the system. However, the point $\mathbf{x} = (0,0)$ fails to satisfy the constraint $x_1 + x_2 \leq -1$ and so does not satisfy the system.

We can now define an Integer Program.

Definition 2.3 *An* Integer Program *consists of a system of integer constraint* $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$ *and vector* $\mathbf{c}$*. The goal is to find a point* $\mathbf{x} \in \mathbb{Z}^n$ *such that* $\mathbf{x}$ *satisfies the system* $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$ *and so that for every other point* $\mathbf{x}'$ *which also satisfies the system* $\mathbf{c} \cdot \mathbf{x} \geq \mathbf{c} \cdot \mathbf{x}'$*. This referred to as a maximization problem and is written*

$$\begin{aligned} \max \mathbf{c} \cdot \mathbf{x} \\ \mathbf{A} \cdot \mathbf{x} \leq \mathbf{b} \end{aligned}$$

Example (3): In the Integer Program

$$\begin{aligned} \max x_1 \\ x_1 + x_2 \leq 2 \\ -x_2 \leq 0 \end{aligned}$$

The optimal solution occurs when $\mathbf{x} = (2,0)$.

From the definition of an integer program we can now define inverse integer programming. In an inverse integer program instead of trying to find a value of $\mathbf{x}$ that maximizes $\mathbf{c} \cdot \mathbf{x}$ we are already given a value for $\mathbf{x}$, say $\mathbf{x}'$, and asked to find a vector $\mathbf{c}$ such that for all $\mathbf{x}$ that satisfy the system $\mathbf{c} \cdot \mathbf{x}' \geq \mathbf{c} \cdot \mathbf{x}$. However without any restrictions on $\mathbf{c}$ this problem can be trivially solved by setting $\mathbf{c} = \mathbf{0}$. Thus we restrict ourselves to finding the vector $\mathbf{c}$ that is closest to a target vector $\mathbf{d}$. The distance between two vectors can be measured in several ways. These distance measures are called norms and in this paper we will focus on two in particular.

Definition 2.4 *The* l_1 *norm of two vectors* $\mathbf{c}$ *and* $\mathbf{d}$ *is* $||\mathbf{c} - \mathbf{d}||_1 = \sum_{i=1}^{n} |c_i - d_i|$

Definition 2.5 *The* l_∞ *norm of two vectors* $\mathbf{c}$ *and* $\mathbf{d}$ *is* $||\mathbf{c} - \mathbf{d}||_\infty = \max_{i=1\ldots n} |c_i - d_i|$

Example (4): Consider the vectors $\mathbf{c} = (1, 2, -4)$ and $\mathbf{d} = (0, -1, -2)$. We have that $||\mathbf{c} - \mathbf{d}||_1 = 1 + 3 + 2 = 6$ and that $||\mathbf{c} - \mathbf{d}||_\infty = 3$.

We can now fully define an Inverse Integer Program.

Definition 2.6 *Given a system of integer constraints, $\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b}$, an integer point $\mathbf{x}'$ such that $\mathbf{A} \cdot \mathbf{x}' \leq \mathbf{b}$, and a target vector d. The Inverse Integer Program is*

$$\min ||\mathbf{c} - \mathbf{d}||$$
$$\forall \mathbf{x} \ \ [\mathbf{A} \cdot \mathbf{x} \leq \mathbf{b} \rightarrow \mathbf{c} \cdot \mathbf{x} \leq \mathbf{c} \cdot \mathbf{x}']$$

Where $||\mathbf{c} - \mathbf{d}||$ is either $||\mathbf{c} - \mathbf{d}||_1$ or $||\mathbf{c} - \mathbf{d}||_\infty$.

The goal of this paper is to show that this problem is Δ_2^P-hard.

3 Motivation

In this section we will mention several ares where Inverse Programming is used. Generally these ares deal with determining what factors are important when trying to predict possible outcomes.

One such area is the field of earthquake prediction. We currently do not know what factors are important in determining when earthquakes will occur or how they will propagate. Inverse optimization can be used to try and extrapolate these important factors given the results of how actual earthquakes occur and propagate.

The parameters involved in modeling earthquakes may be very difficult or impossible to determine. Thus scientists need to estimate these parameters, and values of the observable parameters are used to improve these estimates. This means that inverse problems are extensively studied by geophysical scientists (see, for example, [Neumann84], [Nolet87], [Tarantola87], and [Woodhouse84]).

An important application in this area concerns predicting the movements of earthquakes. To model earthquake movements, consider a network obtained by the discretization of a geologic zone into a number of square cells. Nodes corresponding to adjacent cells are connected by arcs. The cost of an arc represents the transmission time of certain seismic waves from corresponding cells, and is not accurately known. Earthquakes are then observed and the arrival times of the resulting seismic perturbations at various observation stations are observed. Assuming that the earthquakes travel along shortest paths, the problem faced by geologists is to reconstruct the transmission times between cells from the observation of shortest time waves and a priori knowledge of the geologic nature of the zone under study. This problem is an example of an inverse shortest path problem. Inverse problems also arise in X-ray tomography where observations from a CT-scan of a body part together with a priori knowledge of the body is used to estimate its dimension. [Tarantola87] gives a comprehensive treatment of the theory of inverse problems and provides additional applications.

4 Related Work

There have been many papers in the field of inverse optimization. [Zhang96] studied several types of inverse linear programming problems including inverse minimum cost flow and inverse assignment problems. The paper also suggested a method for solving general inverse linear programming problems. This work was extended in [Zhang99]. Then [Yang99] provided two methods for solving inverse linear programs. The first of these used the original LP to provided the initial columns for use by the simplex method. The second is through the ellipsoid method.

[Ahuja01] also studied inverse linear programs. They showed that under the l_1 and l_∞ norms, inverse linear programs are regular linear programs. They also showed that when restricted to unit weights the inverse shortest path and minimum cut problems are still shortest path and minimum cup problems respectively. Several variants of inverse linear programming and methods for solving them were described in [Heuberger04].

A variation of inverse optimization was studied in [Faragó03]. Instead of being given a single original optimization problem, they started with several problems each with a feasible solution and tried to find a single objective function that would make each solution optimal for its original optimization problem. A similar problem was studied in [Roland13], which studied the inverse optimization problem where the goal is to find an an objective function that makes a set of feasible solutions optimal. A further variation was explored in [Dempe06] where the bounds on constraints we allowed to vary along with the objective function.

There have also been several papers on the inverse optimization problems for specific sub-problems of linear programming. [Larin05] studied inverse optimization with regards to periodic linear systems. Also, [Zhang10] studied inverse optimization under the l_1 and l_2 norms of linearly constrained separable programming problems.

Inverse integer programming has also been studied. In [Huang05] the inverse programming problems of several NP-complete problems were studied and shown to be solvable with pseudo-polynomial algorithms. These problems included the knapsack problem and integer programming when restricted to a fixed number of constraints. [Schaefer09] provided two approaches for solving the inverse integer programming problem.

5 Complexity Results

6 Conclusion

In this paper we showed that the inverse integer programming problem is Δ_2^{P}-hard. An interesting research direction would be to study the inverse optimization versions of several variants of integer programming. These include systems of UTVPI constraints, and systems of horn constraints.

References

[Ahuja01] Ahuja, Ravindra K. and Orlin, James B. Inverse Optimization. *Operations Research*, 49(5):771–783, 2001.

[Beck09] Beck, A. and Teboulle, M. A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[Dempe06] Dempe, Stephan and Lohse, Sebastian. Inverse Linear Programming. *Lecture Notes in Economics and Mathematical Systems* 563:19–28, 2006.

[Faragó03] Andrs Faragó and Áron Szentesi and Balázs Szviatovszki Inverse optimization in high-speed networks. *Discrete Applied Mathematics*, 129(1):83–98, 2003.

[Heuberger04] Heuberger, Clemens. Inverse Combinatorial Optimization: A Survey on Problems, Methods, and Results. *Journal of Combinatorial Optimization*, 8(3):329–361, 2004.

[Huang05] Huang, Siming Inverse Problems of Some NP-Complete Problems. *Lecture Notes in Computer Science*, 3521, 422–426, 2005.

[Larin05] Larin, V.B. On the inverse optimization problem for periodic systems. *International Applied Mechanics*, 41(12):1413–1417, 2005.

[Neumann84] Neumann-Denzau, Gertrud and Behrens, Jrn Inversion of seismic data using tomographical reconstruction techniques for investigations of laterally inhomogeneous media. *Geophysical Journal of the Royal Astronomical Society*, 79(1):305–315, 1984.

[Nolet87] Nolet, G. *Seismic Tomography*, Reidel, Dordrecht. 1987.

[Roland13] Julien Roland and Yves De Smet and Jos Rui Figueira Inverse multi-objective combinatorial optimization. *Discrete Applied Mathematics*, 161(16-17):2764–2771, 2013.

[Schaefer09] Schaefer, Andrew J. Inverse integer programming *Optimization Letters*, 3(4):483–489, 2009.

[Tarantola87] Tarantola, A. *Inverse Problem Theory: Methods for Data Fitting and Model Parameter Estimation*. Elsevier, Amsterdam. 1987.

[Woodhouse84] Woodhouse, J. H., and A. M. Dziewonski. Mapping the upper mantle: Three dimensional modeling of Earth structure by inversion of seismic waveforms. *Journal of Geophysical Research*, 89(B7):5953–5986, 1984.

[Yang99] C. Yang and J. Zhang Two general methods for inverse optimization problems. *Applied Mathematics Letters*, 12(2):69–72, 1999.

[Zhang96] Jianzhong Zhang and Zhenhong Liu Calculating some inverse linear programming problems. *Journal of Computational and Applied Mathematics*, 72(2):261–273, 1996.

[Zhang99] Jianzhong Zhang and Zhenhong Liu A further study on inverse linear programming problems. *Journal of Computational and Applied Mathematics*, 106(2):345–359, 1999.

[Zhang10] Jianzhong Zhang and Chengxian Xu Inverse optimization for linearly constrained convex separable programming problems. *European Journal of Operational Research*, 200(3):671–679, 2010.