

SL method for computing a near-optimal solution using linear and non-linear programming in cost-based hypothetical reasoning

M. Ishizuka*, Y. Matsuo

Department of Information and Communication Engineering, School of Engineering, University of Tokyo, 7-3-1 Hongo Bunkyo-ku, Tokyo 113-8656, Japan

Received 2 May 2001; accepted 22 November 2001

Abstract

Hypothetical reasoning is an important framework for knowledge-based systems, however, its inference time grows exponentially with respect to problem size. In this paper, we present an understandable efficient method called *slide-down and lift-up* (SL) method which uses a linear programming technique for determining an initial search point and a non-linear programming technique for efficiently finding a near-optimal 0–1 solution. To escape from trapping into local optima, we have developed a new local handler, which systematically fixes a variable to a locally consistent value. Since the behavior of the SL method is illustrated visually, the simple inference mechanism of the method can be easily understood. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Hypothetical reasoning; Linear programming; Non-linear programming

1. Introduction

By handling incomplete hypothesis knowledge which possibly contradicts with other knowledge, hypothetical reasoning tries to find a set of element hypotheses which is sufficient for proving (or explaining) a given goal (or a given observation) [13]. Because of its theoretical basis and its practical usefulness, hypothetical reasoning is an important framework for knowledge-based systems, particularly for systems based on declarative knowledge. However, since hypothetical reasoning is a form of non-monotonic reasoning and thus an NP-complete or NP-hard problem, its inference time grows exponentially with respect to problem size. In practice, slow inference speed often becomes the most crucial problem.

There already exist several investigations trying to overcome this problem. For example, see Refs. [8–12] for authors' work. Besides symbolic inference methods which have been exploited mostly in AI field, search methods working in continuous-value space have recently shown promising results in achieving efficient inference for hypothetical reasoning as well as for SAT problems and constraint satisfaction problem (CSP). This approach is closely related to mathematical programming, particularly with 0–1 integer programming. When we consider a cost-based propositional hypothetical-reasoning problem, it can

be transformed into an equivalent 0–1 integer programming problem with a set of inequality constraints. Although its computational complexity still remains NP-hard, it allows us to exploit a new efficient search method in continuous-value space. One key point here is an effective use of the efficient simplex method for linear programming, which is formed by relaxing the 0–1 constraint in 0–1 integer programming. Also, non-linear programming formation provides us another possibility. These approaches may be beneficial particularly for developing efficient approximate solution methods, for example, in cost-based hypothetical reasoning [2].

The pivot-and-complement method [1] is a good approximate solution method for 0–1 integer programming. Ishizuka and Okamoto [9] used this method to realize an efficient computation of a near-optimal solution for cost-based hypothetical reasoning. Ohsawa and Ishizuka [12] have transformed the behavior of the pivot-and-complement method into a visible behavior on a knowledge network, improved its efficiency by using the knowledge structure of a given problem, and consequently developed networked bubble propagation (NBP) method. NBP method can empirically achieve a polynomial-time inference of $N^{1.4}$ where N is the number of possible element hypotheses, to produce a good quality near-optimal solution in cost-based hypothetical reasoning.

The key point of these methods is the determination of an initial search point by using simplex method and efficient

* Corresponding author. Tel.: +81-3-5841-6755; fax: +81-3-5841-8570.

local searches around this point in continuous space for eventually finding a near-optimal 0–1 solution. However, in order to avoid trapping into locally optimal points, a sophisticated control of the local search is required; as a result, the inference mechanisms have become complicated and for humans they are difficult to understand.

On the other hand, Gu [5,6] exploited an efficient method to solve SAT problems by transforming them into unconstrained non-linear programming; the SAT problem is related to propositional hypothetical reasoning. There are several methods for unconstrained non-linear programming, i.e. the steepest descent method, Newton's method, quasi-Newton method, and conjugate direction method. The mechanisms of these local search methods are easy to understand as they basically proceed by descending a valley of a function. However, since these methods simply try to find a single solution depending upon an initial search point, they cannot be used for finding a (near) optimal solution for the following reason: if they are trapped into local optima, they have to restart from a new initial point that is to be selected randomly.

In order to find a near-optimal solution using the simple non-linear programming technique, we combine a linear programming, namely simplex method, to determine the initial search point of the non-linear programming. The search, however, often falls into local optima and an effective method escaping from these local optima is required. In this paper, we present an effective method named *variable fixing method* for this problem. Variable fixing corrects a local inconsistency at each locally optimal point and allows to restart the search. Unlike conventional random restart schemes, this method permits to direct the search systematically using the knowledge structure of a given problem.

As sliding-down operations toward a valley of a non-linear function and lifting-up operations are repeated alternately, we call this method *slide-down and lift-up* (SL) method. By illustrating its behavior visually, we show that its mechanism is easily understandable and also achieves good inference efficiency that is close to the efficiency of the NBP method.

In this paper, we will treat hypothetical reasoning problems that are represented in propositional Horn clauses; we will also allow for (in)consistency constraints among hypotheses.

2. Transformation into linear and non-linear programming and their combination

First we show how to transform a hypothetical reasoning problem into linear and non-linear programming problems. These transformations become the basis of the SL method. As for the transformation into linear programming, there are several ways of replacing logical knowledge by an equiva-

lent set of linear inequalities. Among them, we adopt the following transformation used in Ref. [14].

Associating the `true/false` states of logical variables such as $p1, p2, q$, etc. with 1/0 of the corresponding numerical variables represented by the same symbols, we transform a Horn clause

$$q \leftarrow p1 \wedge p2 \quad (1)$$

into a set of inequalities

$$q \leq p1, \quad q \leq p2, \quad p1 + p2 - 1 \leq q \quad (2)$$

and also

$$q \leftarrow p1 \vee p2. \text{ (combination of } q \leftarrow p1 \text{ and } q \leftarrow p2) \quad (3)$$

into

$$p1 \leq q, \quad p2 \leq q, \quad q \leq p1 + p2. \quad (4)$$

This transformation is advantageous in that it allows to produce a 0–1 solution only by using simplex method for a certain type problem [15], though the number of generated inequalities becomes large.

For the constraint representing inconsistency, the head of its Horn clause is set to `false` which is translated to 0 in the corresponding inequality. As the goal of hypothetical reasoning has to be satisfied, it is set to `true` which becomes 1 in the inequality.

Let the weights of possible element hypotheses $h1, h2, h3, \dots$ be w_1, w_2, w_3, \dots , respectively. Moreover, let the element hypothesis hi ($i = 1, 2, \dots$) become $hi = 1$ if it is included in the solution hypothesis, and $hi = 0$ otherwise. Then we can define the cost of the solution as

$$\text{cost} = w_1h1 + w_2h2 + w_3h3 + \dots$$

which expresses the sum of the weights of the element hypotheses included in the solution. Let us set cost as the objective function; then if we compute the optimal solution to minimize this function under the generated inequalities, it indicates the optimal solution in the cost-based hypothetical reasoning problem.

In this way, hypothetical reasoning becomes a 0–1 linear integer programming. In the pivot-and-complement method (Balas 80), an efficient approximate solution method for 0–1 integer programming, the optimal real-number solution is obtained as follows: first the simplex method is used by relaxing the 0–1 constraint on the variables, then a near-optimal 0–1 solution is searched in a sophisticated manner around the optimal real-number solution. This local search mechanism for the 0–1 solution is rather complicated because it incorporates several heuristics that have been obtained empirically. For our new method here, while we utilize this optimal real-number solution obtained by the simplex method as the initial search point, we try to develop a new simple and understandable method of the local search using a non-linear programming technique.

Gu [5,6] presented a method for SAT problems by transforming them into unconstrained non-linear programming

متن کامل مقاله

دریافت فوری ←

ISIArticles

مرجع مقالات تخصصی ایران

- ✓ امکان دانلود نسخه تمام متن مقالات انگلیسی
- ✓ امکان دانلود نسخه ترجمه شده مقالات
- ✓ پذیرش سفارش ترجمه تخصصی
- ✓ امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
- ✓ امکان دانلود رایگان ۲ صفحه اول هر مقاله
- ✓ امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
- ✓ دانلود فوری مقاله پس از پرداخت آنلاین
- ✓ پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات