Journal of Global Optimization (2006) 36: 219–236
DOI 10.1007/s10898-006-9007-2

Analyzing the Complexity of Finding Good Neighborhood Functions for Local Search Algorithms*

DEREK E. ARMSTRONG[1] and SHELDON H. JACOBSON[2]
[1]*Los Alamos National Laboratory, Los Alamos, NM 87544*
[2]*Department of Mechanical and Industrial Engineering University of Illinois at Urbana-Champaign Urbana, IL 61801-2906 (e-mail: shj@uiuc.edu)*

(Received 18 September 2002; accepted in revised form 31 January 2006; Published online 14 June 2006)

Abstract. A drawback to using local search algorithms to address NP-hard discrete optimization problems is that many neighborhood functions have an exponential number of local optima that are not global optima (termed L-locals). A neighborhood function η is said to be *stable* if the number of L-locals is polynomial. A stable neighborhood function ensures that the number of L-locals does not grow too large as the instance size increases and results in improved performance for many local search algorithms. This paper studies the complexity of stable neighborhood functions for NP-hard discrete optimization problems by introducing *neighborhood transformations*. Neighborhood transformations between discrete optimization problems consist of a transformation of problem instances and a corresponding transformation of solutions that preserves the ordering imposed by the objective function values. In this paper, MAX Weighted Boolean SAT (MWBS), MAX Clause Weighted SAT (MCWS), and Zero-One Integer Programming (ZOIP) are shown to be NPO-complete with respect to neighborhood transformations. Therefore, if MWBS, MCWS, or ZOIP has a stable neighborhood function, then every problem in NPO has a stable neighborhood function. These results demonstrate the difficulty of finding effective neighborhood functions for NP-hard discrete optimization problems.

Subject Classification: analysis of algorithms, computational complexity

Key words: Computational complexity, Local search algorithms, NP-hard discrete optimization problems

1. Introduction

Local search algorithms (Aarts and Lenstra, 1997) are used to address hard discrete optimization problems. An instance I of a discrete optimization problem can be denoted as a two-tuple (SOL(I), m), where SOL(I) is a countable solution space and m: SOL(I)$\rightarrow Q$ is the objective function. A local search algorithm is a procedure that iteratively moves between

* This research is supported in part by the Air Force Office of Scientific Research (F49620-01-1-0007, FA9550-04-1-0110).

solutions in search of an optimal or near-optimal solution of a discrete optimization problem. A local search algorithm can be defined by a neighborhood function and an acceptance probability. For every instance I of a discrete optimization problem, a neighborhood function $\eta(I, .)$: $\mathrm{SOL}(I) \rightarrow 2^{\mathrm{SOL}(I)}$ defines the movement through the solution space, where the current solution must be a neighbor of the previous solution. An acceptance probability defines the probability that a neighboring solution is accepted to be the next solution. The choice of neighborhood function can have a tremendous impact on the effectiveness of a local search algorithm.

One consideration when choosing a neighborhood function is the number of local optima that are not global optima induced by the neighborhood function. Given a neighborhood function η and an instance I of discrete maximization problem, a solution $s \in \mathrm{SOL}(I)$ is a (*strict*) *local optimum* if $f(s)(>) \geqslant f(s')$ for all $s' \in \eta(I, s)$, and a solution $s \in \mathrm{SOL}(I)$ is a *global optimum* if $f(s) \geqslant f(s')$ for all $s' \in \mathrm{SOL}(I)$. Similar definitions hold for a discrete minimization problem. A solution s is a (strict) *L-local* if s is a (strict) local optimum that is not a global optimum. Local search algorithms often become trapped at L-locals and are unable to continue to find improving solutions. Many local search algorithms will be more effective if the neighborhood function has fewer L-locals.

A drawback of using local search algorithms for NP-hard discrete optimization problems is that many computationally efficient neighborhood functions have a very large number of L-locals. For a given solution, a *polynomially computable neighborhood* can be searched in polynomial time for an improving solution or else it is concluded that the current solution is a local optimum. It is necessary for a neighborhood function to be polynomially computable to ensure that iterations of the local search algorithm can each be completed in polynomial time. Based on the experiences of numerous researchers, one can conjecture that most (and possibly all) polynomially computable neighborhood functions for NP-hard discrete optimization problems have exponentially many L-locals (Rodl and Tovey, 1987; Armstrong and Jacobson, 2003; Armstrong and Jacobson, 2005). Given a discrete optimization problem A, a neighborhood function η for A is said to be *stable* if η is computable in polynomial time with the property that the number of L-locals is bounded above by a polynomial in the length of problem instances. A stable neighborhood function ensures that the number of L-locals does not grow too large as the instance size increases.

This paper studies the complexity of finding effective neighborhood functions for several NP-hard discrete optimization problems. New types of reductions between discrete optimization problems are introduced that consist of a transformation of instances between the problems and a corresponding transformation of solutions that preserves the ordering imposed by the objective function values. The new reductions facilitate a method to study the complexity of neighborhood functions with certain desirable properties. Order transformations between discrete optimization problems are defined to preserve the objective function values over problems' solution spaces. Order transformations are defined such that the following holds: if problem A order transforms to problem B, then for every instance I of A there exists an instance $f(I)$ (which can be constructed in polynomial time in the length of I) of B such that the number of global optima in I equals the number of global optima in $f(I)$, the number of solutions with the second best objective function value for I equals the number of solutions with the second best objective function value for $f(I)$, and so on, up to the worst solution value for the instance I. There are few results in the literature regarding transformations between discrete optimization problems that preserve the ordering imposed by the objective function. Ausiello et al. (1980) define a reduction between convex discrete optimization problems that preserves the ordering imposed by the objective function.

A restricted version of order transformations is introduced, termed neighborhood transformations. Neighborhood transformations are used to study the complexity of polynomially computable neighborhood functions with a limited number of L-locals. It is shown that if A neighborhood transforms to B and B has a stable neighborhood function, then A has a stable neighborhood function. Also, if A neighborhood transforms to B and B has a polynomially computable neighborhood function in which the number of L-locals is bounded above by a fixed integer k ($\geqslant 0$), then A has a polynomially computable neighborhood function in which the number of L-locals is bounded above by k. Hence, the neighborhood transformation also preserves inclusion in PGS (Jacobson and Solow, 1993); i.e., if A neighborhood transforms to B and $B \in$ PGS, then $A \in$ PGS. Informally, the class of problems PGS is defined to be those discrete optimization problems for which there exists a polynomially computable neighborhood function with zero L-locals. The neighborhood transformation also preserves the property of a unique global maximum (or minimum), which means that if problem A neighborhood transforms to a problem B, then every instance I of A with a unique global maximum can be mapped (in polynomial time) to an instance $f(I)$ of problem B with a unique global maximum. As a result of this property, it may be possible to use the neighborhood transformation to study the complexity of optimization problems with unique global optima (note that this is not addressed in this paper), which has been addressed by Pardalos and Jha (1992) and Prokopyev et al. (2005), who prove that quadratic 0–1 programming and hyperbolic 0–1 programming, respectively, are NP-hard even when instances are restricted to have unique global optima.

Several problems are shown to be NPO-complete with respect to order transformations and neighborhood transformations. Problem $B \in$NPO is said to be NPO-complete with respect to a transformation, denoted by

$\propto$, if for every problem $A \in$ NPO, $A \propto B$. Generally, complete problems form a class of the hardest problems with respect to a certain property. In particular, the NPO-complete problems studied in this paper will be the hardest discrete optimization problems in terms of formulating stable neighborhood functions and polynomial time improvement algorithms. In this paper, MAX Weighted Boolean SAT (MWBS), MAX Clause Weighted SAT (MCWS), and Zero-One Integer Programming (ZOIP) are all shown to be NPO-complete with respect to neighborhood transformations; these three problems will be formally defined in Section 2. Armstrong (2002) also shows that several other problems, such as MAX Weighted Independent Set (MWIS) and MAX Weighted Clique (MWC), are NPO-complete with respect to neighborhood transformations. Therefore, if any one of MWBS, MCWS, ZOIP, MWIS, or MWC has a stable neighborhood function, then every problem in NPO has a stable neighborhood function.

The research on approximation preserving reductions (see Ausiello et al., 1995 for a survey) has some similarities with the work presented in this paper. The set of problems APX consist of all problems A in NPO for which there exists a polynomial time r-approximate algorithm (Ausiello et al., 1999) for some $r \geqslant 1$. Ausiello and Protasi (1995) introduce a new class of optimization problems called Guaranteed Local Optima (GLO). A maximization (minimization) problem is in GLO if there is a constant h such that the value of all local optima, with respect to some defined neighborhood function, is at least $1/h$ (at most h) times the value of global optima. The set of optimization problems GLO are compared to the set of optimization problems APX by Ausiello and Protasi (1995). The set of problems PTAS (polynomial time approximation scheme) contains all problems in NPO for which there exist a polynomial time r-approximate algorithm for all $r > 1$. Several reductions between discrete optimization problems have been defined to preserve inclusion in APX or PTAS. For example, Crescenzi and Trevisan (2000) introduce a new polynomial time approximation scheme that preserves approximations, called PTAS-reducibility, which generalizes other transformations that preserve approximations. They show that if A PTAS-reduces to B and $B \in$ PTAS, then $A \in$ PTAS. Crescenzi and Trevisan (2000) also show that MAX-SAT is APX-complete under the PTAS-reducibility. Some problems have been shown to be NPO-complete with respect to an approximation preserving reduction. Ausiello et al. (1995) show that MWBS is NPO-complete with respect to an approximation preserving reduction. It should be noted that all of the approximation preserving reductions given in the literature do not preserve order. Also, order transformations defined in this paper do not preserve approximability.

An open research question is to determine if PGS equals P_S or, otherwise, is it possible for an optimization problem to be in PGS and not solvable in polynomial time (note that $P_S \subseteq$PGS). A limited number of papers (Grotschel and Lovasz, 1995; Schulz et al., 1995; Schulz and Weismantel, 1999) report results showing that for particular NP-hard discrete optimization problems, being in PGS is sufficient for the problem to be polynomially solvable. The class of problems PLS (Johnson et al., 1988) is similar to the class PGS in that both classes contain problems where solutions can be improved in polynomial time until the "goal" is reached (where the "goal" for PGS is reaching a global optimum and the goal for PLS is reaching a local optimum). Johnson et al. (1988) introduce the class PLS to study the complexity of finding *local optima* for a neighborhood function of an NP-hard problem (e.g., can local minima of the 2-change neighborhood for the traveling salesman problem be found in polynomial time?). The results in this paper differs from those reported in Johnson et al. (1988) in that neighborhood transformations are used to analyze the complexity of finding *neighborhood functions* for NP-hard problems with certain properties (e.g., does the traveling salesman problem have a stable neighborhood function?).

In many cases, the average-case performance of local search algorithms has been very good (Hoos and Stutzle, 2004). The work in this paper consists of a worst-case analysis and thus, does not contradict the extensive amount of literature on the effective average-case performance of local search. Furthermore, if a NP-hard discrete optimization problem does not have any stable neighborhood functions, this does not necessarily imply that all local search algorithms will perform poorly. Local search algorithms that use random-restart techniques and allow hill climbing moves (i.e., moves to solutions with worse objective function value) are designed to help mitigate the problems associated with exponentially many local optima. The purpose of this paper is to develop a better understanding of the properties of neighborhood functions of NP-hard discrete optimization problems. An ultimate goal is to discover the properties of neighborhood functions that are inherent to NP-hard discrete optimization problems.

The paper is organized as follows: Section 2 provides formal definitions and background material needed to develop the main results. Order transformations, neighborhood transformations, and several discrete optimization problems are formally defined in Section 2. Section 3 presents fundamental theoretical results and properties of the order transformation. In particular, theoretical results are given that demonstrate the properties preserved by the order transformation. Section 4 provides results about the neighborhood transformation and several discrete optimization problems are shown to be NPO-complete with respect to neighborhood transformations. Section 5 provides concluding comments and directions for future research.

2. Definitions and Background

To describe the results, several definitions are needed. An NPO problem will be formally defined since the notation needed to define it will be used throughout the remainder of the paper. An NPO problem A is a four-tuple (D_A, SOL, m, goal) such that:

(1) D_A is the set of instances of A and it is recognizable in polynomial time in the length of problem instances.
(2) Given an instance $I \in D_A$, SOL(I) denotes the set of feasible solutions of I. Furthermore, there exists a polynomial function p such that for any $s \in$ SOL(I), $|s| \leqslant p(|I|)$. The solutions can also be recognized in time polynomial in the length of I.
(3) For each instance I and solution $s \in$ SOL(I), $m(I, s)$ denotes the measure or objective function value of solution s. The function m is also computable in time polynomial in the length of I.
(4) goal $\in \{\max, \min\}$ represents whether the problem is a minimization or maximization problem.

Let NPO$^+$ denote the set of NPO problems in which, for instances I, a solution $s \in$ SOL(I) can be generated in polynomial time in the length of I. For every problem $A = (D_A$, SOL, m, goal) in NPO, there exists a corresponding nondeterministic Turing machine N_A such that for every instance $I \in D_A$, N_A executes the following algorithm in polynomial time:

Algorithm 1 (Ausiello et al., 1995):
begin
guess a solution $s \in \{0, 1\}^{p(|I|)}$;
if $s \in$ SOL(I) then output $m(I, s)$; else abort
end

This algorithm guesses a solution and computes the objective function value of any feasible solution. This algorithm is an abstract definition that is used in a proof of a theorem in Section 4.

For every instance I of a discrete optimization problem $A = (D_A$, SOL, m, goal), a *neighborhood function* $\eta(I, s) \subseteq$ SOL(I) maps each solution $s \in$ SOL(I) into a subset of the solution space. The *order* of a solution s is the rank of the solution in terms of the objective function values. For example, if I is an instance of a maximization problem $A = (D_A$, SOL, m, goal) in which m(SOL(I)) = $\{2, 4, 7, 9, 12\}$, then any solution s such that $m(I, s) = 12$ will have an order of one, any solution s with $m(I, s) = 9$ has an order of two, and so on, such that any solution s with $m(I, s) = 2$ has an order of five.

The purpose of this paper is to develop transformations between discrete optimization problems that preserve the local structure or ordering imposed by the objective function. These are called *order transformations*.

A neighborhood transformation, which is a restricted version of an order transformation, preserves polynomially computable neighborhood functions from one discrete optimization problem to another discrete optimization problem. Throughout the remainder of this paper, it is assumed that the discrete optimization problems are maximization problems. Therefore, any problem in NPO can be represented by a three-tuple (D_A, SOL, m). Let A and B be two discrete maximization problems in NPO. The problem $A = (D_A,\ \mathrm{SOL}_A,\ m_A)$ *order transforms* to $B = (D_B,\ \mathrm{SOL}_B,\ m_B)$, if there exists two computable functions f and q such that the following holds:

(1) For each instance I of A, $f(I)$ is an instance of B and f is computable in polynomial time in the length of I.
(2) For each instance I of A and for each $s \in \mathrm{SOL}_A(I)$, $q(I, s) \in \mathrm{SOL}_B(f(I))$. For simplicity, denote $q(I, s)$ by $q(s)$. For each fixed I, the function $q(I, .)$ is one-to-one and satisfies the following properties:
   (a) it can be determined (in polynomial time in the length of instance I) if a solution $s_B \in \mathrm{SOL}_B(f(I))$ is also in $q(\mathrm{SOL}_A(I))$,
   (b) for every instance I of A and for every s, $s' \in \mathrm{SOL}_A(I)$, if $m_A(I, s)(>) \geqslant m_A(I, s')$, then $m_B(f(I),\ q(s))(>) \geqslant m_B(f(I), q(s'))$.
   (c) for every $s \in q(\mathrm{SOL}_A(I))$ and $s' \in \mathrm{SOL}_B(f(I)) - q(\mathrm{SOL}_A(I))$, $m_B(f(I), s) \geqslant m_B(f(I), s')$.

The purpose of condition 2c is to ensure that for each instance I, the set of solutions that are not in $q(\mathrm{SOL}_A(I))$ have an objective function value no larger than any solution in $q(\mathrm{SOL}_A(I))$. This condition (in relation to the other conditions) forces the local structure of instance $f(I)$ to be the same as the instance I, except that the instance $f(I)$ can have more solutions than the instance I. If the function q and its inverse are computable in polynomial time in the length of I, the transformation is said to be a *neighborhood transformation*. The order transformation and neighborhood transformation are both transitive.

Several discrete optimization problems that are used in the paper will now be described. For the rest of this paper, let zero (0) denote a false literal or clause and one (1) denote a true literal or clause.

MAX SAT: Given m clauses, over n Boolean variables $X = \{x_1,\ x_2, \ldots,\ x_n\}$, find a truth assignment $t: X \to \{0, 1\}$ that maximizes the number of satisfied clauses.

Two related discrete optimization problems are now described.
MAX Clause Weighted SAT (MCWS): Given a set of clauses $C = \{c_1,\ c_2, \ldots.,\ c_m\}$, with corresponding weights $W = \{w_1, w_2, \ldots, w_m\}$, over the Boolean variables $X = \{x_1, x_2, \ldots, x_n\}$, find a truth assignment $t: X \to \{0, 1\}$ that maximizes the sum of the weights of the satisfied clauses.

To define MAX Weighted Boolean SAT, the following definition is needed. A *Boolean formula* is an expression that can be constructed from a set of Boolean variables $x_1, x_2, \ldots, x_n$ and using $\wedge$ (AND), $\vee$ (OR), and $\sim$ (NOT) operators.

MAX Weighted Boolean SAT (MWBS): Given a Boolean formula F over the Boolean variables $X = \{x_1, x_2, \ldots, x_n\}$ with corresponding weights $W = \{w_1, w_2, \ldots, w_n\}$, find a truth assignment $t : X \to \{0, 1\}$ that satisfies the Boolean formula F such that $\sum_{i=1}^{n} w_i t(x_i)$ is maximized.

Given an instance I of MWBS, the set of solutions SOL(I) consists of all truth assignments that satisfy F. Note that MAX Clause Weighted 3-SAT (MCW3S) is a particular case of MCWS in which every clause in C has exactly three literals, and MAX Weighted Boolean 3-SAT (MWB3S) is a particular case of MWBS in which the Boolean formula F is in 3-conjunctive normal form (3-CNF).

Two other discrete optimization problems are now defined.

(0–1) Knapsack: Given vectors $s = (s_1, s_2, \ldots, s_n), c = (c_1, c_2, \ldots, c_n)$, where $s_i,\ c_i \in Z^+$, and capacity $B \in Z^+$, find a vector $\boldsymbol{x} = (x_1,\ x_2, \ldots, x_n) \in \{0, 1\}^n$ that maximizes $\sum_{i=1}^{n} c_i x_i$ subject to $\sum_{i=1}^{n} s_i x_i \leqslant B$.

Zero-One Integer Programming (ZOIP): Maximize $\sum_{i=1}^{n}\{c_i x_i\}$ subject to $\sum_{j=1}^{n} a_{ij} x_j \leqslant b_i$, for $i = 1, 2, \ldots, m$, where $X = (x_1, x_2, \ldots, x_n) \in \{0, 1\}^n$, and the c_i, a_{ij} are integers.

3. Fundamental Results Regarding Order Transformations

In this section, fundamental results regarding the order transformations will be given. Suppose A neighborhood transforms to B. The following result is noted and can be used to develop a general result given by Proposition 1: for every polynomially computable neighborhood function for B there exists a corresponding polynomially computable neighborhood function for A such that for every instance I of A, there is a polynomially computable instance $f(I)$ of B, where the number of L-locals and strict L-locals for instance I is bounded above by the number of L-locals and strict L-locals, respectively, for instance $f(I)$ of B. Also, sufficient conditions for the existence of a neighborhood transformation from one problem to another problem will be given. One of these sufficient conditions will be used in Section 4 to show that the (0-1) knapsack neighborhood transforms to another NP-hard discrete optimization problem.

Suppose $A = (D_A,\ \mathrm{SOL}_A, m_A)$ neighborhood transforms to $B = (D_B,\ \mathrm{SOL}_B, m_B)$ such that f and q denote the transformation of instances of A to instances of B and the corresponding transformation of solutions, respectively. Given a neighborhood function η for B, define a neighborhood function $\eta_{f,q}$ for A as follows:

$$\text{for all } I \in D_A \text{ and } s \in \text{SOL}_A(I), \eta_{f,q}(I,s) = \{s' : q(s') \in \eta(f(I), q(s))\}.$$

Since the functions f, q, and q^{-1} are all computable in polynomial time, if η can be computed in polynomial time, then $\eta_{f,q}$ can be computed in polynomial time for A.

Proposition 1 provides a general result on how polynomially computable neighborhood functions are preserved by a neighborhood transformation from one discrete optimization problem to another problem. The proofs of all propositions in this section follow directly from the definition of neighborhood transformations (Armstrong, 2002) and are, thus, omitted.

PROPOSITION 1. *Let A and B be two discrete optimization problems in NPO such that A neighborhood transforms to B. Furthermore, let $p_2: Z^+ \rightarrow Z^+$ be a non-decreasing function. If B has a polynomially computable neighborhood function such that the number of (strict) L-locals is bounded above by $p_2(|I_2|)$ for each instance I_2 of B, then A has a polynomially computable neighborhood function such that the number of (strict) L-locals is bounded above by $p_2(p_1(|I_1|))$ for each instance I_1 of A, where p_1 is some polynomial function.*

The result in Proposition 1 for neighborhood transformations will also hold for local optima and strict local optima in place of L-locals and strict L-locals. These results are not explicitly given since the difficulty in addressing a discrete optimization problem with local search algorithms arises out of the number of local optima that are not global optima rather than the number of local optima (which also includes global optima). Note that Proposition 1 implies that if there exists a stable neighborhood function for B, then A has a stable neighborhood function. Similarly, other results can be obtained from Proposition 1. For example, if there exists a polynomially computable neighborhood function for B that has no more than k L-locals, then A has a polynomially computable neighborhood function with no more than k L-locals.

Propositions 2 and 3 delineate properties of a neighborhood transformation by noting the preservation of a polynomial time improvement algorithm and the preservation of a polynomial time algorithm to find a solution of a specified order or better.

PROPOSITION 2. *Let A and B be two discrete optimization problems in NPO such that A neighborhood transforms to B. If $B \in$ PGS, then $A \in$ PGS.*

By Proposition 2, neighborhood transformations can be used to study the complexity of polynomial time improvement of non-optimal solutions.

PROPOSITION 3. *Let $A \in NPO^+$ and $B \in NPO$ be two discrete optimization problems such that A neighborhood transforms to B. If there exists a polynomial time algorithm that can find a solution with order k, or less, for instances of B, then there exists a polynomial time algorithm to find a solution with order k, or less, for instances of A.*

The next two propositions provide sufficient conditions for the existence of a neighborhood transformation between two discrete optimization problems.

PROPOSITION 4. *Let $A = (D_A,\ SOL_A,\ m_A)$ and $B = (D_A,\ SOL_A, m_B)$ be two discrete optimization problems in NPO, where for every $I \in D_A$ and $s \in SOL_A(I)$, $m_B(I,s) = a(I)m_A(I,s) + b(I)$ such that $a > 0$ and b are functions mapping instances of A to Z. Then A neighborhood transforms to B.*

PROPOSITION 5. *Suppose $\varphi : R \rightarrow R$ is an increasing function. Let $A = (D_A,\ SOL_A,\ m_A)$ and $B = (D_A,\ SOL_A,\ m_B)$ be two discrete optimization problems in NPO, where for every $I \in D_A$ and $s \in SOL_A(I)$, $m_B(I,s) = \varphi(m_A(I,s)) + b(I)$. Then A neighborhood transforms to B.*

Proposition 4 states that there exists an order transformation between discrete optimization problems when their objective functions are linearly related. In the next two sections, it is shown that there exist neighborhood transformations between problems that appear to have dissimilar objective functions. Moreover, several problems are shown to be NPO-complete with respect to neighborhood transformations.

4. Transformations that Preserve Local Structure

In this section, by using Proposition 4, an example will be given of a neighborhood transformation from one discrete optimization problem to another discrete optimization problem. Furthermore, MWBS, MCWS, and ZOIP will be shown to be NPO-complete with respect to neighborhood transformations. From the propositions in Section 3, the following statements will then hold: If there exists a polynomial time algorithm that can find a solution with order k, or less, of instances of MWBS, MCWS, or ZOIP, then for every problem A in NPO^+, there must exist a polynomial time algorithm to find a solution of order k, or less, for instances of A. Also, if there exists a stable neighborhood function for MWBS, MCWS, or ZOIP, then every problem in NPO has a stable neighborhood function. Lastly, if any of the problems MWBS, MCWS, or ZOIP is in PGS, then every problem in NPO is also in PGS. Proposition 6 uses Proposition 4 to provide an example of a neighborhood transformation from (0–1) knapsack to another NP-hard discrete optimization problem.

PROPOSITION 6. *Let $k \geqslant 1$ be an integer. Let A denote the discrete optimization problem:*

Maximize $\sum_{i=1}^{n} c_i x_i^2$ subject to $\sum_{i=1}^{n} s_i x_i \leqslant B, k \leqslant x_i \leqslant k+1$, where $s=(s_1, s_2, \dots, s_n)$, $c=(c_1, c_2, \dots, c_n)$, $s_i, c_i \in Z^+$, and $B \in Z^+$. Then, (0–1) Knapsack neighborhood transforms to A.

Proof. Suppose I is an instance of (0–1) Knapsack with n possible items. Let the instance I be denoted by $s=(s_1, s_2, \dots, s_n)$, $c=(c_1, c_2, \dots, c_n)$, where s_i, $c_i \in Z^+$, and integer B. Define the instance $f(I)$ of A by using the same parameters s, c, and B. Therefore, the instance $f(I)$ consists of n independent variables. The solution space of instance $f(I)$ can be represented in the same manner as the solution space for instance I of problem A. Define the function q: $\{0,1\}^n \to \{k, k+1\}^n$ such that for each $x \in \{0,1\}^n, q(x)=k\mathbf{1}+x$ where $\mathbf{1}=(1,1,\dots,1) \in Z^n$. Given a solution $y \in \{k, k+1\}^n$ for instance $f(I)$ of A, the objective function value for y is given by $\sum_{i=1}^{n} c_i (x_i+k)^2$, where $x_i = y_i - k$.

Since $(x_i)^2 = x_i$, then

$$\sum_{i=1}^{n} c_i (x_i+k)^2 = \sum_{i=1}^{n} c_i (k^2 + x_i^2 + 2kx_i) = \sum_{i=1}^{n} c_i k^2 + (2k+1) \sum_{i=1}^{n} c_i x_i$$

Therefore, the objective function for the instance $f(I)$ can be written as a linear combination, $a(I) m_{\mathrm{KNAP}}(I, .) + b(I)$, of the objective function m_{KNAP} for the instance I of (0–1) Knapsack, where $a(I) > 0$. By Proposition 4, this implies that (0–1) Knapsack neighborhood transforms to A. □

The neighborhood transformation used in the proof of Proposition 6 is between two similar discrete optimization problems. The restrictive nature of neighborhood transformations does not prevent the development of completeness results. Theorem 1 shows that MWBS is NPO-complete with respect to neighborhood transformations by adapting a proof in Ausiello et al. (1995).

THEOREM 1. *MWBS is NPO-complete with respect to neighborhood transformations.*

Proof. Let $A = (D, \text{SOL}, m_A)$ be any problem in NPO. Without loss of generality suppose that $m_A(I, s) \geqslant 0$ for all $I \in D$ and $s \in SOL(I)$. The proof will be complete by showing that A neighborhood transforms to

MWBS. Let N_A denote the nondeterministic Turing machine corresponding to the discrete optimization problem A, as described in Section 2. Let I be an instance of A. By using Cook's Theorem (Cook, 1971), there exists a Boolean formula φ_I such that there is a one-to-one correspondence between the satisfying truth assignments of φ_I and the halting computation paths of $N_A(I)$. Now, let $s_1, s_2, \ldots, s_p$ be the Boolean variables that describe a solution $s \in \mathrm{SOL}(I)$. Also, let $m_1, m_2, \ldots, m_l$ denote the Boolean variables on which $N_A(I)$ writes the value $m_A(I, s)$. Let q_Y and q_N be the Boolean variables that represent N_A halting in a "yes" ($s \in \mathrm{SOL}(I)$) state and "no" ($s \notin \mathrm{SOL}(I)$) state, respectively. Note that the Boolean variables m_i, s_j, q_Y, and q_N will be part of the formula φ_I. For the Boolean variable m_i assign the weight 2^{l-i}. For the Boolean variable q_Y assign the weight 2^l. Assign all the remaining Boolean variables a weight of zero. The instance $f(I)$ of MWBS is given by the Boolean formula φ_I and the corresponding Boolean variables together with the weights specified above. Given a solution $s \in \mathrm{SOL}(I)$ that results in a halting computation, define $q(s)$ to be the unique satisfying truth assignment for φ_I that corresponds to s (which can be obtained by executing $N_A(I)$ on the input s). It then follows that the objective function value of the solution $q(s)$ in this created instance $f(I)$ of MWBS is exactly $m_A(I, s) + 2^l$. Given a satisfying truth assignment t of the Boolean formula φ_I, the solution $q^{-1}(s)$ is obtained by simply noting the value of $t(s_i)$ for all I. Therefore, the function q^{-1} is computable in time polynomial in the length of I. Note that any satisfying truth assignment t for the instance $f(I)$ in which $t(q_Y) = 1$ will have an objective function value of at least 2^l. Also, any truth assignment t' in which $t'(q_Y) = 0$ will have an objective function value that is less than 2^l. This implies that A neighborhood transforms to MWBS. □

Theorem 2 states that MWB3S is NPO-complete with respect to neighborhood transformations. The proof of this theorem is omitted (see Armstrong, 2002) since it follows from Theorem 1 and a natural extension of the polynomial transformation from SAT to 3-SAT (Garey and Johnson 1979).

THEOREM 2. *MWB3S is NPO-complete with respect to neighborhood transformations.*

Theorem 3 shows that MCW3S is NPO-complete with respect to neighborhood transformations by proving that MWB3S neighborhood transforms to MCW3S.

THEOREM 3. *MCW3S is NPO-complete with respect to neighborhood transformations.*

Proof. The proof follows by showing that MWB3S neighborhood transforms to MCW3S. Let C be a set of clauses in which each clause has exactly three literals over the Boolean variables $X = \{x_1, x_2, \ldots, x_n\}$, with weights w_i for each $x_i \in X$; this defines an instance I of MWB3S. Let $X' = X \cup \{(y_{i1}, y_{i2}): 1 \leqslant i \leqslant N\}$. Define the two sets of clauses $\bar{C} = \{(x_i, y_{i1}, y_{i2}): 1 \leqslant i \leqslant n\}$ and $\hat{C} = \{(\bar{x}_i, \bar{y}_{i1}, \bar{y}_{i2}), (\bar{x}_i, y_{i1}, \bar{y}_{i2}), (\bar{x}_i, \bar{y}_{i1}, y_{i2}), (x_i, \bar{y}_{i1}, \bar{y}_{i2}), (x_i, y_{i1}, \bar{y}_{i2}), (x_i, \bar{y}_{i1}, y_{i2}) : 1 \leqslant i \leqslant n\}$. Let $C' = C \cup \bar{C} \cup \hat{C}, m = |C \cup \hat{C}|$, and $M = \sum_{i=1}^{n} w_i$. Define an instance $f(I)$ of MCW3S by using the set of clauses C' over the set of Boolean variables X' . The weights corresponding to the clauses in C' are now specified. Let each clause in C and $\hat{C}$ have a weight of M. For each clause (x_i, y_{i1}, y_{i2}) in $\bar{C}$, assign a weight of w_i. Define the function q as follows: for each satisfying truth assignment $t: X \to \{0, 1\}$, let $q(t) = \bar{t}$ where $\bar{t}: X' \to \{0, 1\}, \bar{t}(x_i) = t(x_i)$ for all $x_i \in X$, and $\bar{t}(y) = 0$ for all $y \in X' - X$. By construction, for any truth assignment $t: X \to \{0, 1\}$ that satisfies all of the clauses in C, $m_{\text{MCW3S}}(f(I), q(t)) = Mm + \sum_{i=1}^{n} w_i t(x_i)$. Therefore, for any two truth assignments $t: X \to \{0, 1\}$ and $t': X \to \{0, 1\}$ that satisfy C, if $\sum_{i=1}^{n} w_i t(x_i)(>) \geqslant \sum_{i=1}^{n} w_i t'(x_i)$, then $m_B(f(I), q(t))(>) \geqslant m_B(f(I), q(t'))$. Now, suppose that $\bar{t}: X' \to \{0, 1\} \in q(\text{SOL}(I))$ and $\bar{t}': X' \to \{0, 1\} \notin q(\text{SOL}(I))$. Since $\bar{t}' \notin q(\text{SOL}(I))$, then there exists $y \in X' - X$ such that $\bar{t}'(y) = 1$ or $\bar{t}'$ does not satisfy C. In either case, it follows that one of the clauses in $C \cup \hat{C}$ cannot be satisfied. This implies that $m_{\text{MCW3S}}(f(I), \bar{t}') \leqslant M(m-1) + \sum_{i=1}^{n} w_i = Mm$. Therefore, $m_B(f(I), \bar{t}') \leqslant Mm \leqslant m_B(f(I), \bar{t})$. The transformation f of the instances and the corresponding transformation q of the solutions can be executed in polynomial time in the length of I. Furthermore, the solutions in $q(\text{SOL}(I))$ can be recognized in polynomial time in the length of I, hence MWB3S neighborhood transforms to MCW3S. By the transitivity of neighborhood transformations and Theorem 2, MCW3S is NPO-complete with respect to neighborhood transformations. □

COROLLARY 1. *MCWS is NPO-complete.*

Proof. MCW3S is a special case of MCWS, hence the result directly follows from Theorem 3. □

Theorem 4 shows that ZOIP is NPO-complete with respect to neighborhood transformations by proving that MWB3S neighborhood transforms to ZOIP.

THEOREM 4. *ZOIP is NPO-complete with respect to neighborhood transformations.*

Proof. The proof follows by showing that MWB3S neighborhood transforms to ZOIP. Start with an instance I of MWB3S, where this instance consists of a set of clauses $C = \{c_1, c_2, \ldots, c_m\}$, where each clause in C has exactly three literals, over a set of Boolean variables $X = \{x_1, x_2, \ldots, x_n\}$

with a weight w_i associated with each Boolean variable $x_i \in X$. An instance $f(I)$ of ZOIP is created which preserves the ordering imposed by the objective function. Every Boolean variable x_i will have a corresponding Boolean variable in $f(I)$, with this Boolean variable denoted using the same notation. For each clause $c_j = (y_{1j}, y_{2j}, y_{3j})$, the constraint

$$h(y_{1j}) + h(y_{2j}) + h(y_{3j}) \geqslant 1,$$
$$\text{where } h(y_{ij}) = \begin{cases} 1 - x_k & \text{if } y_{ij} \text{ is a negated literal of variable } x_k \\ x_k & \text{if } y_{ij} \text{ is a nonnegated literal of variable } x_k \end{cases},$$

is included in the instance $f(I)$. For example, if the clause $(x_2, \bar{x}_5, x_6)$ is in C, then the constraint $x_2 + (1 - x_5) + x_6 \geqslant 1$ is part of the instance $f(I)$. Also, each Boolean variable x_i will have the constraint that $x_i = 0$ or 1. Finally, the objective function is $\sum_{i=1}^{n} w_i x_i$. Let $X = (x_1, x_2, \ldots, x_n) \in \{0, 1\}^n$ be a solution that satisfies all of the clauses in C. By construction, x satisfies all of the constraints for instance $f(I)$. Therefore, letting $q(s) = s$ for all $s \in \mathrm{SOL}(I)$, it then follows that $m_{MWB3S}(I, s) = m_{ZOIP}(f(I), q(s))$. Hence, MAX Weighted 3-SAT neighborhood transforms to ZOIP. $\square$

Armstrong (2002) shows that many other problems are NPO-complete with respect to neighborhood transformations. Theorem 5 shows that there exists a large class of discrete optimization problems such that every problem in that class neighborhood transforms to MAX SAT. The following definition from Ausiello et al. (1995) is needed to prove this result.

DEFINITION 1. (Ausiello et al. 1995), Let a Σ_n-*formula* (respectively, a Π_n-*formula*) be a prefix first-order formula with N alternating blocks of quantifiers beginning with $\exists$ (respectively, $\forall$). The class MAX-Σ_n (respectively, MAX-Π_n) consists of maximization problems A whose instances and solutions are finite structures I and S, respectively, and the optimum measure on input I is definable by the expression

$$m^*(I) = \max_{S} |\{x : \varphi(x, S, I)\}|,$$

where φ is a Σ_n-formula (respectively, a Π_n-formula) and x is a tuple of fixed dimension whose components range over the domain of I.

The first part of the proof of Theorem 5 is adapted from a proof in Papadimitriou and Yannakakis (1991). They prove that every problem in MAX-Σ_0 L-reduces to MAX SAT. The L-reduction is a restricted type of transformation between optimization problems, defined for addressing approximability issues. Their proof is extended to show that A neighborhood transforms to MAX SAT for all problems A in MAX-Σ_0. Therefore, the proof given in Papadimitriou and Yannakakis (1991) is a nice example of a transformation that preserves approximability but does not preserve local structure.

THEOREM 5. *For $A \in MAX\text{-}\Sigma_0$, A neighborhood transforms to MAX SAT.*

Proof. Let $A \in$ MAX-Σ_0. From the definition of MAX-Σ_0, the objective function value may be written as $m(s) = |\{x: \varphi(x, s, I)\}|$, where φ is a quantifier free first order formula, x is a tuple of fixed dimension whose components range over the domain of I, and S is the structure corresponding to a feasible solution. Suppose I is an instance of A. For instance I, let $\varphi_1, \varphi_2, \dots, \varphi_m$ be formulae that correspond to the possible values of x. Therefore, the problem of maximizing the set of values x that satisfy φ is reduced to the problem of maximizing the set of formulae φ_i that can be satisfied by a truth assignment. Each formula φ_i will be transformed into a set of clauses C_i. Consider the Boolean circuit corresponding to the formula φ_i (where the inputs are the original variables of φ_i) and represent all of the gates by clauses over a set of Boolean variables that contains the original variables of φ_i. The representation of the gates for formula φ_i will be as follows:

(1) If g is a NOT gate with input a, C_i includes the clauses (g, a) and $(\bar{g}, \bar{a})$.
(2) If g is a AND gate with inputs a and b, C_i includes the clauses $(\bar{g}, a)$, $(\bar{g}, b)$, and $(g, \bar{a}, \bar{b})$.
(3) If g is a OR gate with inputs a and b, C_i includes the clauses $(g, \bar{a})$, $(g, \bar{b})$, and $(\bar{g}, a, b)$.
(4) If g is the output gate, C_i includes the clause (g).

This transformation is designed in such a way that any truth assignment over the Boolean variables that make up the clauses in C_i can be extended to satisfy all but one clause (the clause corresponding to the output gate) of C_i. Let $C' = C_1 \cup C_2 \cup \cdots \cup C_m$ and X' denote the set of Boolean variables over which C' is defined. Let M denote the number of clauses in C'. Therefore, given a solution S for instance I with value $m - k$, for some nonnegative integer k, a unique truth assignment for X' can be constructed such that all but k clauses of C' are satisfied.

It remains to show that A neighborhood transforms to MAX SAT. Let i be an integer such that $1 \leqslant i \leqslant m$ and define $C_i' = \{(c_j, z_{ip}) : c_j \in C_i$ and c_j is not the output gate of C_i, $1 \leqslant p \leqslant m\} \cup \{(\bar{z}_{ip}) : 1 \leqslant p \leqslant m\} \cup \{(\bar{z}_{ip}, z_{il}) : 1 \leqslant p,\ l \leqslant m, p \neq l\}$, where the $'z'$ variables are additional variables needed for the neighborhood transformation. The instance $f(I)$ of MAX SAT is given by $C'' = C' \cup C_1' \cup C_2' \cup \cdots \cup C_m'$ and $X'' = X' \cup \{z_{ip} : 1 \leqslant i, p \leqslant m\}$. Let S be a feasible solution for instance I such that $m(S) = m - k$ for some nonnegative integer k. Define $q_1(S)$ to be the corresponding truth assignment $t: X' \to \{0, 1\}$ that satisfies all clauses in $m - k$ of the sets C_1,

$C_2, \ldots, C_m$ and satisfies all but one clause in the remaining k sets. Now, for each truth assignment $t: X' \to \{0, 1\}$, define $q_2(t)$ to be the truth assignment $t'': X'' \to \{0, 1\}$ such that $t''(x) = t(x)$ for all $x \in X'$ and $t''(x) = 0$ for all $x \in X'' - X'$. The transformation of solutions in the definition of a neighborhood transformation is given by $q_2(q_1(S))$ for all solutions S of instance I. Given two solutions S_1 and S_2 for the instance I such that $m(S_1) \geqslant m(S_2)$, then $q_2(q_1(S_1))$ must satisfy at least as many clauses as $q_2(q_1(S_2))$. Also, given two solutions S_1 and S_2 for the instance I such that $m(S_1) > m(S_2)$, then $q_2(q_1(S_1))$ satisfies more clauses than $q_2(q_1(S_2))$. Note that for any solution S for the instance I, $q_2(q_1(S))$ satisfies at least $|C''| - m$ clauses.

Given a truth assignment $t'': X'' \to \{0, 1\}$ such that $t''(x) = 1$, for some $x \in X'' - X$, then at least m clauses of C'' are not satisfied. Moreover, any truth assignment t'' that does not satisfy every clause in C_i (except possibly the clause corresponding to the output gate of C_i, for $i = 1, 2, \ldots, m$) will not satisfy at least m clauses of C''. Therefore, it follows that the number of clauses satisfied by any truth assignment in $q_2(q_1(\mathrm{SOL}(I))$ is at least as many as the number satisfied by any truth assignment not in $q_2(q_1(\mathrm{SOL}(I))$ $\equiv q$. The function q and its inverse are computable in polynomial time in the length of instance I. Therefore, A neighborhood transforms to MAX SAT. □

5. Conclusion and Future Directions

This paper analyzed the complexity of effective neighborhood functions for NP-hard discrete optimization problems. To this end, the concept of an order transformation was introduced. Order transformations preserve the exact structure from one discrete optimization problem to another discrete optimization problem in terms of the ordering of the objective function values. A restricted type of order transformation, called a neighborhood transformation, is introduced which preserves polynomially computable neighborhood functions between discrete optimization problems. This paper shows that there are NPO-complete problems with respect to neighborhood transformations. It was shown that MWBS, MCWS, and ZOIP are all NPO-complete with respect to neighborhood transformations. These completeness results establish the difficulty in finding stable neighborhood functions for these NP-hard discrete optimization problems. The completeness results also demonstrate that it is very unlikely for these optimization problems to have a polynomially computable neighborhood function with a limited number of L-locals. Using the propositions in Section 3 together with the completeness results, the following results are verified:

(1) If there exists a stable neighborhood function for one of the problems MWBS, MCWS, or ZOIP, then every problem in NPO has a stable neighborhood function.
(2) If there exists a polynomially computable neighborhood function for one of MWBS, MCWS, or ZOIP, in which the number of L-locals is bounded above by k, then every problem in NPO has a polynomially computable neighborhood function in which the number of L-locals is bounded above by k.
(3) If there exists a polynomial time algorithm to find the k^{th} best solution (or better) for instances of one of MWBS, MCWS, or ZOIP, then every problem in NPO^+ has a polynomial time algorithm to find the k^{th} best solution (or better).
(4) If MWBS, MCWS, or ZOIP is in PGS, then every problem in NPO is in PGS.

These results suggest several other research directions. It would be useful to find other problems that are complete with respect to neighborhood transformations (Armstrong, 2002). In particular, it is desirable to show that every problem in a large set of NPO problems, in which the corresponding decision problem is NP-complete, is NPO-complete with respect to neighborhood transformations. This result may be difficult to obtain due to the restricted nature of neighborhood transformations. Therefore, another goal is to determine what properties of NP-hard discrete optimization problems indicate that it will be NPO-complete with respect to neighborhood transformations. Research is in progress to study the characteristics of polynomially computable neighborhood functions for NP-hard discrete optimization problems. The overall purpose of this research is to gain a better understanding of the properties of NP-hard discrete optimization problems in terms of local search algorithm characteristics.

References

Aarts, E. and Lenstra, J.K. (eds.) (1997), *Local Search in Combinatorial Optimization*, John Wiley & Sons, Chichester.

Armstrong, D.E. (2002), A local search algorithm approach to analyzing the complexity of discrete optimization problems, Ph.D. Dissertation, University of Illinois, Urbana, IL.

Armstrong, D.E. and Jacobson, S.H. (2005), Data independent neighborhood functions and strict local optima, *Discrete Applied Mathematics* 143, 272–284.

Armstrong, D.E. and Jacobson, S.H. (2003), Studying the complexity of global verification for NP-hard discrete optimization problems, *Journal of Global Optimization* 27, 83–96.

Ausiello, G., D'Atri A. and Protasi, M. (1980), Structure preserving reductions among convex optimization problems, *Journal of Computer and System Sciences* 21, 136–153.

Ausiello, G., Crescenzi P. and Protasi, M. (1995), Approximate solution of NP optimization problems, *Theoretical Computer Science* 150, 1–55.

Ausiello, G., Crescenzi, P., Gambosi, G., Kann, Marchetti-Spaccamela, A. and Protasi, M. (1999), *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*, Springer-Verlag, Berlin.

Ausiello, G. and Protasi, M. (1995), Local search, reducibility and approximability of NP-optimization problems, *Information Processing Letters* 54, 73–79.

Cook, S.A. (1971), The complexity of theorem-proving procedures, *Proceedings of 3rd ACM Symposium on Theory of Computation*, 151–158.

Crescenzi, P. and Trevisan, L. (2000), On approximation scheme preserving reducibility and its applications, *Theory of Computing Systems* 33, 1–16.

Garey, M.R. and Johnson, D.S. (1979), *Computers and Intractability: A Guide to the Theory of NP-Completeness*, W.H. Freeman and Company, New York, NY.

Grotschel, M. and Lovasz, L. (1995), Combinatorial optimization, In: Graham, R., Grotschel, M. and Lovasz, L. (eds.), *Handbook of Combinatorics*, vol. II. pp. 1541–1597. North-Holland, Amsterdam.

Hoos, H.H. and Stutzle T. (2004), *Stochastic Local Search: Foundations and Applications*, Morgan Kaufmann, San Francisco.

Jacobson, S.H. and Solow, D. (1993), The effectiveness of finite improvement algorithms for finding global optima, *Methods and Models of Operations Research* 37, 257–272.

Johnson, D.S., Papadimitriou, C.H. and Yannakakis, M. (1988), How Easy is Local Search? *Journal of Computer and System Sciences* 37, 79–100.

Papadimitriou, C.H. and Yannakakis, M. (1991), Optimization, approximation, and complexity classes, *Journal of Computer and System Sciences* 43, 425–440.

Pardalos, P.M. and Jha, S. (1992), Complexity of uniqueness and local search in quadratic 0–1 programming, *Operations Research Letters* 11, 119–123.

Prokopyev, O.A., Huang, H. and Pardalos P.M. (2005), On complexity of unconstrained hyperbolic 0–1 programming problems, *Operations Research Letters* 33, 312–318.

Rodl, V. and Tovey, C.A. (1987), Multiple optima in local search. *Journal of Algorithms* 8, 250–259.

Schulz, A.S. and Weismantel, R. (1999), An oracle-polynomial time augmentation algorithm for integer programming, *Proceedings of the 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, Baltimore, MD, 967–968.

Schulz, A.S., Weismantel, R. and Ziegler, G.M. (1995), 0/1-Integer programming: optimization and augmentation are equivalent, In: Spirakis, P. (ed.), *Lecture Notes in Computer Science* 979, pp. 473–483. Springer, Berlin, Germany.