Ameso Optimization: a Relaxation of Discrete Midpoint Convexity

Wen Chen

Providence Business School, Providence College, Providence, RI 02908

Odysseas Kanavetas

Department of Management Science and Information Systems, Rutgers Business School, Newark and New Brunswick, 180 University Avenue, Newark, NJ 07102-1895

Michael N. Katehakis

Department of Management Science and Information Systems, Rutgers Business School, Newark and New Brunswick, 180 University Avenue, Newark, NJ 07102-1895

Abstract

In this paper we introduce a special class of discrete optimization problems the ameso programming problems. We show that for the one dimension ameso optimization problems there are simple, to verify, optimality conditions at any optimal point. Further we construct a procedure that can solve multi-dimensional ameso optimization problems without necessarily performing complete enumeration.

Keywords: Discrete Optimization, Integral Optimization, Reduction Procedure

1. Introduction

In this paper we introduce a new class of discrete optimization problems the ameso programming problems. For the one dimension case we have shown that any optimal point can be determined by simple, to verify, optimality conditions. Furthermore, we have constructed the Ameso Reduction Procedure (ARP) that solves ameso optimization problems without necessarily performing complete enumeration. Parallel implementations of the ARP can easily be done, c.f. [1]. Since this is a new class of problems there is no directly related literature. However, since an ameso problem can be a generalization and relaxation of midpoint convexity there are algorithms proposed in other papers that employ the proximity framework while using descent algorithms for discrete midpoint convex functions, c.f. [11]. Also, based on
the midpoint convexity there are many other approaches to nonlinear integer optimization as in [9] and other more algebraic methods have been developed in the last two decades, as described in [4], [6], [12]. Finally, one can find a lot of applications where proving that a model is ameso we can have very simple algorithm to obtain the optimal solution, as described in [18].

The rest of the paper is organized as follows. In section 2, we define the ameso optimization problem and we discuss its properties. In section 2.1 there is the relationship between convexity, midpoint convexity and ameso optimization problem. Section 2.2 is devoted to the one dimensional case and in section 2.3 we discuss the main property of the high dimensional case and we use this property to design a procedure to solve ameso(C) optimization problems. Also, we present two examples that illustrate the performance of the proposed procedure. Finally, in the conclusion we discuss the relaxation that ameso(1) provides to the midpoint convexity and potential benefits.

2. ameso(C) Optimization Problem

Given a subset $D^n$ of the $n$-dimensional integers, $D^n \subseteq \mathbb{Z}^n$, and a real function $f$ defined on $D^n$, we define the following:

**Definition 1.** $D^n$ is called an **ameso set**, if it satisfies the following condition
\[
\left\lfloor \frac{x + y}{2} \right\rfloor, \left\lfloor \frac{x + y}{2} \right\rfloor \in D^n, \text{ for all } x, y \in D^n.
\]

**Definition 2.** $(D^n, f)$ is called an **ameso(C) pair**, if and only if it satisfies the following conditions
- the domain $D^n$ of the function $f(\cdot)$ is an ameso set,
- $f(\cdot)$ has a lower bound and
- there exists $C \geq 0$ such that the following holds for all $x, y \in D^n$
\[
f(x) + f(y) + C \geq f\left(\left\lfloor \frac{x + y}{2} \right\rfloor\right) + f\left(\left\lfloor \frac{x + y}{2} \right\rfloor\right).
\]

**Definition 3.** Minimization of $f(x)$ subject to $x \in D^n$ is an **ameso(C) optimization problem**, if $(D^n, f)$ is an ameso(C) pair.

**Notation.** For notational simplicity in the sequel for any integers $a$ and $b$ we will use notation $[a, b]$, $[a, b)$, and $(a, b]$ to denote respectively the sets of integers: $\{a, a+1, \ldots, b\}$, $\{a, a+1, \ldots, b-1\}$, and $\{a+1, \ldots, b\}$.

For better understanding of the definition of an ameso set we provide some examples below. Also, note that for notational simplicity we use $D^n$ to denote sets that are subsets of $\mathbb{Z}^n$ that are not necessarily products of identical subsets of $\mathbb{Z}$. This is demonstrated in the Example 1 below.
Example 1: Given any integers $a_i, b_i$ where $a_i < b_i$, if we let $D_i = [a_i, b_i]$, $i = 1, ..., n$, then $D^n = D_1 \times \cdots \times D_n$ is an ameso set.

Example 2: The sets
\[ A_1 = \{(x_1, x_2) : x_2 - x_1 \geq 3, x_2 \leq 10, x_1, x_2 \in \mathbb{Z}^+\}, \]
\[ A_2 = \{(x_1, x_2) : x_2 - x_1 \geq 0, x_2 \leq 10, x_1 \leq 5, x_1, x_2 \in \mathbb{Z}^+\}, \]
\[ A_3 = \{(x_1, x_2) : x_2 - x_1 \leq 10, x_1, x_2 \in \mathbb{Z}^+\}, \]
are ameso sets. However, the sets
\[ A_4 = \{(x_1, x_2) : 3x_2 - x_1 \geq 0, x_1, x_2 \in \mathbb{Z}^+\}, \]
\[ A_5 = \{(x_1, x_2) : x_1 + 2x_2 \geq 10, x_1, x_2 \in \mathbb{Z}^+\}, \]
are not ameso sets since the points $(3, 1), (12, 4) \in A_4$, but $(7, 2) \notin A_4$, and also points $(8, 1), (2, 4) \in A_5$, but $(5, 2) \notin A_5$.

Example 3: For $D^1 = [-20, 20]$ and $f(x) = \frac{1}{4}x^4 - x^3 + x$, $x \in D^1$, it is easy to see that $D^1$ is an ameso set and $f(x) + f(y) + 4 \geq f\left(\frac{x+y}{2}\right) + f\left(\frac{x+y}{2}\right)$ for every $x, y \in D^1$. Thus $(D^1, f)$ is an ameso pair, i.e., $C = 4$. And the minimization problem of $f(x)$ subject to $x \in D^1$ is an ameso optimization.

Below we state some properties which follow from the definition of an ameso(C) pair. The second and third properties can be used for ameso relaxation of complicated functions where it is difficult to obtain directly that they are an ameso pair.

**Property 1.** If $(D^n, f)$ is an ameso(C) pair then the following inequality holds for all $\bar{x} + \bar{a}, \bar{x} - \bar{a} \in D^n$

\[ f(\bar{x} + \bar{a}) + f(\bar{x} - \bar{a}) + C \geq 2f(\bar{x}). \]

**Property 2.** If $(D^n, f)$ is an ameso(C1) pair and $C_2 > C_1$, then $(D^n, f)$ is an ameso(C2) pair.

**Property 3.** If $(D^n, f)$ is an ameso(C) pair, and $(D^n, g)$ is an ameso(C') pair, then $(D^n, af + bg)$, $a, b \geq 0$, is an ameso(aC + bC') pair.

2.1. Relation Between ameso Optimization and Convexity

In this section we prove that if the domain of a convex function is an ameso set and if $f$ is a bounded discrete midpoint convex function, then $(\mathbb{Z}^n, f)$ is an ameso pair. This result points to how useful the ameso optimization framework can be for discrete optimization problems.

To start recall the definition of a convex function $f$ in one dimension,

\[ \lambda f(x) + (1 - \lambda)f(y) \geq f(\lambda x + (1 - \lambda)y), \quad x, y \in \mathbb{R}. \]

Next we state the relation of a convex function on an ameso set and an ameso pair.

**Proposition 1.** Let $f_i(x)$, $x \in \mathbb{R}$, is a convex function and there exists a lower bound for each $i$, $1 \leq i \leq n$. For a function $g(\bar{y}) = \sum_{i=1}^{n} a_i f_i(y_i)$, $a_i \geq 0$, the $(\mathbb{Z}^n, g)$ is an ameso(0) pair.
Proof: Clearly $\mathbb{Z}^n$ is an ameso set and $g$ has a lower bound. For every $i, 1 \leq i \leq n$ and $x_i, y_i \in \mathbb{Z}$, we will show that

$$f_i(x_i) + f_i(y_i) \geq f_i\left(\left\lfloor \frac{x_i + y_i}{2} \right\rfloor \right) + f_i\left(\left\lceil \frac{x_i + y_i}{2} \right\rceil \right).$$

(1)

For the case $\frac{\bar{x} + \bar{y}}{2} \in \mathbb{Z}$, we have that $\left\lfloor \frac{x_i + y_i}{2} \right\rfloor = \left\lceil \frac{x_i + y_i}{2} \right\rceil = \frac{x_i + y_i}{2}$. Hence, we have

$$f_i(x_i) + f_i(y_i) \geq 2f_i\left(\frac{x_i + y_i}{2}\right) = f_i\left(\left\lfloor \frac{x_i + y_i}{2} \right\rfloor \right) + f_i\left(\left\lceil \frac{x_i + y_i}{2} \right\rceil \right).$$

The inequality comes from the convexity of $f_i$.

Now, if $\frac{\bar{x} + \bar{y}}{2} \notin \mathbb{Z}$, then clearly $x_i \neq y_i$. Let $z^a = \min\{x_i, y_i\}$ and $z^b = \max\{x_i, y_i\}$. Then, the following three statements are true

$$z^a - 1 < z^a < z^a + 1 \leq z^b,$$

$$z^b - \frac{z^a + z^b}{2} = \frac{z^b - z^a}{2} = \frac{z^b - z^a}{2} \geq 0,$$

$$\frac{z^a + z^b}{2} = \left\lfloor \frac{z^a + z^b}{2} \right\rfloor = \frac{z^a + z^b}{2} + 0.5 = \left\lceil \frac{z^a + z^b}{2} \right\rceil + 1.$$

Using the convexity of $f_i$ for the above points, we have that

$$f_i(z^b) - f_i\left(\frac{z^a + z^b}{2}\right) \geq f_i\left(\frac{z^a - 1 + z^b}{2}\right) - f_i(z^a),$$

$$f_i(z^a) + f_i(z^b) \geq f_i\left(\frac{z^a + z^b}{2}\right) + f_i\left(\left\lceil \frac{z^a + z^b}{2} \right\rceil \right).$$

Also, since $z^a = \min\{x_i, y_i\}$ and $z^b = \max\{x_i, y_i\}$, we have

$$f_i(x_i) + f_i(y_i) = f_i(z^a) + f_i(z^b) \geq f_i\left(\left\lfloor \frac{z^a + z^b}{2} \right\rfloor \right) + f_i\left(\left\lceil \frac{z^a + z^b}{2} \right\rceil \right) = f_i\left(\left\lfloor \frac{x_i + y_i}{2} \right\rfloor \right) + f_i\left(\left\lceil \frac{x_i + y_i}{2} \right\rceil \right).$$

Therefore, we have proved that (1) holds for every $i$ and every $x_i, y_i \in \mathbb{Z}$. Hence, we have

$$g(\bar{x}) + g(\bar{y}) = \sum_{i=1}^{n} a_i f_i(x_i) + \sum_{i=1}^{n} a_i f_i(y_i) = \sum_{i=1}^{n} a_i (f_i(x_i) + f_i(y_i))$$

$$\geq \sum_{i=1}^{n} a_i (f_i\left(\left\lfloor \frac{x_i + y_i}{2} \right\rfloor \right) + f_i\left(\left\lceil \frac{x_i + y_i}{2} \right\rceil \right))$$

$$= g\left(\left\lfloor \frac{x_i + y_i}{2} \right\rfloor \right) + g\left(\left\lceil \frac{x_i + y_i}{2} \right\rceil \right).$$

Thus, $(\mathbb{Z}^n, g)$ is an ameso(0) pair. □

Now, from the above Proposition and by recalling the definition of a discrete midpoint convex function

$$f(\bar{x}) + f(\bar{y}) \geq f\left(\left\lfloor \frac{\bar{x} + \bar{y}}{2} \right\rfloor \right) + f\left(\left\lceil \frac{\bar{x} + \bar{y}}{2} \right\rceil \right), \quad \forall x, y \in \mathbb{Z}^n,$$

we obtain the following property.

**Property 4.** If $f$ is a bounded discrete midpoint convex function, then $(\mathbb{Z}^n, f)$ is an ameso(0) pair.
2.2. Properties of the one-dimensional ameso(C) Pair

In this section we state and prove properties of the one dimensional ameso optimization. The optimization algorithm we propose in this paper is a decomposition algorithm i.e., it is based on finding solutions to simpler one dimensional problems, see also [2], [3]. We start with the following lemma which shows us the form of an ameso set. According to this lemma an ameso set can be expressed only in the form $[a, b]$.

**Lemma 1.** An one-dimension set $M \subset \mathbb{Z}$ is an ameso set if and only if it can be expressed as $[x_s, x_t]$, with $x_s < x_t$.

**Proof:** First consider a set $M = [x_s, x_t]$. Then for all $x, y \in M$, we have that $x, y \in [x_s, x_t]$ and $x, y$ are integers. Also, it is easy to show that $min\{x, y\} \leq \lfloor \frac{x+y}{2} \rfloor \leq \lceil \frac{x+y}{2} \rceil \leq max\{x, y\}$. Therefore, $[\frac{x+y}{2}], [\frac{x+y}{2}] \in M$. That is, $M$ is an one-dimension ameso set.

Conversely, we will prove that every one-dimension ameso set, $D^1$, can be expressed as $[x_s, x_t]$ where $x_s < x_t$ are integers.

First by relabeling we can write $D^1 = \{a_1, a_2, ..., a_m\}$ where $a_1 < a_2 < \ldots < a_m$, where $a_i - a_i - 1 \geq 1$.

We next show that $a_i - a_i - 1 = 1$. To prove this claim assume that there exists $i$, $a_i - a_i - 1 \geq 2$, for this $i$ we have $\lfloor \frac{a_i + a_i - 1}{2} \rfloor = \lceil \frac{a_i - 1 + 1}{2} \rceil = a_i - 1 + 1$ and $\lfloor \frac{a_i + a_i - 1}{2} \rfloor \leq \lceil \frac{2a_i - 2}{2} \rceil = a_i - 1 = a_i - 1$. It follows that $a_i - 1 < \lfloor \frac{a_i + a_i - 1}{2} \rfloor < a_i$ and $\lfloor \frac{a_i + a_i - 1}{2} \rfloor$ is not in the set $D^1$, by its construction. This contradicts the definition of the ameso set and the proof of the claim is complete.

Since for all $i$, $a_i - a_i - 1 = 1$ then $D^1$ can be expressed by $[a_1, a_m]$ and the proof is complete. □.

To avoid trivial cases in the sequel, we assume that the ameso sets under study are not empty or singletons. Further according to Lemma 1, $D^1 = [x_s, x_t]$, where $x_s, x_t$ are unique integers and $x_s < x_t$, for simplicity this later interval $[x_s, x_t]$, will be denoted by $I_0(= I_0(D^1))$.

Next we discuss properties of the one-dimension ameso(C) pair $(D^1, f)$, where we use the notation: $D^1 = [x_s, x_t] = I_0$. In the following lemmas, we prove that if there exists an interval $[a, b]$ in the domain $[x_s, x_t]$ where we can prove that $a$ is the minimum of function $f$ on the interval $[a, b]$ under some conditions then it is the minimum of $f$ on the interval $[a, x_t]$.

**Lemma 2.** If there exist $x^0 \in D^1$ and $b \in \mathbb{Z}^+$ such that $[x^0, x^0 + b] \subseteq I_0$ and the following conditions hold

a) $f(x^0) = min_{y \in [x^0, x^0 + b]} f(y)$,

b) $f(x^0) + C \leq max_{y \in [x^0, x^0 + b]} f(y)$,

then

$\max\{w : f(w) = max_{y \in [x^0, x^0 + b]} f(y)\} \in (x^0 + \frac{b}{2}, x^0 + b]$. 

5
Proof: Let $S_1 = [x^0, x^0 + b]$, and let $z = \max \{w : f(w) = \max_{y \in [x^0,x^0+b]} f(y)\}$, then $S_1 \subseteq I_0$ and we need to prove that $z \in (x^0 + \frac{b}{2}, x^0 + b)$. Also, we have from the definition of $z$ that $f(x^0) + C \leq f(z)$.

If $C = 0$, we have that for all $x$, $f(x+1) + f(x-1) \geq 2f(x)$, because of Property 1. That is, $f(x+1) - f(x) \geq f(x) - f(x-1)$, for all $x$. Since $f(x^0) = \min_{y \in S_1} f(y)$, and that $f$ is increasing in this interval we have that $x_0 + b$ is the maximum of $f$ on this interval. Thus, $\max \{w : f(w) = \max_{y \in S_1} f(y)\} = x^0 + b$ for the case $C = 0$.

Now we assume that $C > 0$. Because of $f(x^0) + C \leq f(z)$, it is easy to show that $z \neq x^0$. Assume $z \in (x^0, x^0 + \frac{b}{2}]$. We have $2z - x^0 \in (x^0, x^0 + b] \Rightarrow 2z - x^0 \in S_1$ and

$$2z - x^0 = z + z - x^0 > z + 0 = z \Rightarrow 2z - x^0 > z. \quad (2)$$

Because $z = \max \{w : f(w) = \max_{y \in S_1} f(y)\}$,

$$f(2z - x^0) < f(z). \quad (3)$$

Because $(D^1, f)$ is an ameso$(C)$ pair, we have $f(x^0) + f(2z - x^0) + C \geq 2f(z)$, then $f(x^0) + f(z) + C > 2f(z)$ because of (3). Therefore, $f(x^0) + C > f(z)$. But this statement contradicts with $f(x^0) + C \leq f(z)$. So $z$ can not be in $(x^0, x^0 + \frac{b}{2}]$, at the same time $z \neq x^0$. In other words, $\max \{w : f(w) = \max_{y \in S_1} f(y)\} \in (x^0 + \frac{b}{2}, x^0 + b]$ for the case $C > 0$.

Summarizing, $\max \{w : f(w) = \max_{y \in S_1} f(y)\} \in (x^0 + \frac{b}{2}, x^0 + b]$. □

**Lemma 3.** If there exist $x^0 \in D^1$ and $b \in \mathbb{Z}^+$ such that $[x^0, x^0 + b] \subseteq I_0$ and the following conditions hold

$$f(x^0) = \min_{y \in [x^0,x^0+b]} f(y),$$

$$f(x^0) + C \leq \max_{y \in [x^0,x^0+b]} f(y),$$

then $f(x^0) = \min_{y \in [x^0,x^0+b]} f(y)$.

**Proof:** Let $S_1 = [x^0, x^0 + b]$, $I_+ = [x^0, x^0]$, then $S_1 \subseteq I_0$, $I_+ \subseteq I_0$, $f(x^0) = \min_{y \in S_1} f(y)$, and $f(x^0) + C \leq \max_{y \in S_1} f(y)$.

Now, if $x^0 + b = x_t$, then $S_1 = I_+$. The result follows from Lemma 2.

For $x^0 + b < x_t$, we will use mathematical induction to prove it. Let $S_1' = [x^0, x^0 + b + 1]$. Since, $S_1' \supset S_1$, we have that $\max_{y \in S_1'} f(y) \geq \max_{y \in S_1} f(y)$. Also, from the condition $f(x^0) + C \leq \max_{y \in S_1} f(y)$, it follows that $f(x^0) + C \leq \max_{y \in S_1'} f(y)$.

Now, we will show that $f(x^0) = \min_{y \in S_1} f(y)$. From Lemma 2, under the condition

$$f(x^0) + C \leq \max_{y \in S_1} f(y),$$
we have that there exists
\[
\max \{ w : f(w) = \max_{y \in S_1} f(y) \} \in (x_0 + \frac{b}{2}, x_0 + b].
\]

Let \( z = \max \{ w : f(w) = \max_{y \in S_1} f(y) \} \). Now, it is easy to show that \( 2z \in (2x_0 + b, 2x_0 + 2b] \) and \( 2z - (x_0 + b + 1) \in (x_0 - 1, x_0 + b - 1] \). Since \( 2z - (x_0 + b + 1) \) is integer, we have that \( 2z - (x_0 + b + 1) > x_0 - 1 \Leftrightarrow 2z - (x_0 + b + 1) \geq x_0 \). That is, \( 2z - (x_0 + b + 1) \in [x_0, x_0 + b - 1]. \) Therefore,
\[
f(2z - (x_0 + b + 1)) \leq f(z)
\] (4)

Now, since \((D^1, f)\) is an ameso(C) pair, \( f(x_0+b+1) + f(2z-(x_0+b+1)) + C \geq 2f(z) \). Using 2 that means \( f(x_0+b+1) + f(z) + C \geq 2f(z) \). Hence, \( f(x_0+b+1) \geq f(z) - C \geq f(x_0) \). That is, \( f(x_0) \leq f(x_0+b+1) \).

Also, \( S'_1 = (S_1) \cup \{x_0 + b + 1\} \) and \( f(x_0) = \min_{y \in S_1} f(y) \). Therefore, \( \min_{y \in S'_1} f(y) = f(x_0) \).

We have shown that for the interval \( S'_1 = [x_0, x_0 + b + 1] \), both \( f(x_0) = \min_{y \in S'_1} f(y) \) and \( f(x_0) + C \leq \max_{y \in S'_1} f(y) \) hold.

Similarly, let \( S''_1 = [x_0, x_0 + b + 2] \), we also can prove that both \( f(x_0) = \min_{y \in S''_1} f(y) \) and \( f(x_0) + C \leq \max_{y \in S''_1} f(y) \) hold. Therefore, with mathematical induction we have shown that
\[
f(x_0) = \min_{y \in [x_0, x_t]} f(y)
\]
and
\[
f(x_0) + C \leq \max_{y \in [x_0, x_t]} f(y),
\]
hold. This completes the proof. \( \square \).

The following theorem shows that if there exists an interval \([a, b]\) in the domain and where \( a \), the minimum of \( f \) in this interval, is also minimum for the interval \([a, x_t]\) then this local minimum is global minimum, i.e., minimum in \([x_s, x_t]\).

**Theorem 1.** If there exist \( x_0 \in D^1 \) and \( b \in \mathbb{Z}^+ \) such that \([x_0, x_0 + b] \subseteq I_0 \) and the following conditions hold
\[
f(x_0) = \min_{y \in [x_s, x_0+b]} f(y),
\]
\[
f(x_0) + C \leq \max_{y \in [x_0, x_0+b]} f(y),
\]
then \( f(x_0) = \min_{y \in D^1} f(y) \).

**Proof:** Let \( S_1 = [x_0, x_0 + b], I_+ = [x_0, x_t], I_- = [x_s, x_0], \) thus \( f(x_0) = \min_{y \in (I_- \cup S_1)} f(y) \), and \( f(x_0) + C \leq \max_{y \in S_1} f(y) \). Since \( f(x_0) = \min_{y \in (I_- \cup S_1)} f(y) \) and \( x_0 \in I_-, x_0 \in S_1 \), then \( f(x_0) = \min_{y \in I_-} f(y) \), and \( f(x_0) = \min_{y \in S_1} f(y). \) Now, from the condition \( f(x_0) + C \leq \max_{y \in S_1} f(y) \) and according to Lemma 3, we have that \( f(x_0) = \min_{y \in I_+} f(y) \). Also, \( f(x_0) = \min_{y \in I_-} f(y) \), \( I_0 = I_+ \cup I_- \), \( D^1 = I_0 \cup \mathbb{Z} \). Therefore \( f(x_0) = \min_{y \in D^1} f(y) \). \( \square \)

Below we give a corollary which provides a way to find an interval in the domain with the properties we derived above.
Corollary 1. If there exist \( x' \in D^1, z \in (x', x_t] \) with \( f(z) - f(x') \geq C \), then

\[
\min_{y \in [x, z]} f(y) = \min_{y \in D^1} f(y).
\]

**Proof:** Let \( f(x^0) = \min_{y \in [x, z]} f(y) \), we need to prove that \( \min_{y \in D^1} f(y) = f(x^0) \).

Let \( f(\hat{x}) = \min_{y \in [\hat{x}, z]} f(y) \), then \( f(\hat{x}) = \min_{y \in [\hat{x}, z]} f(y) \). Therefore, \( f(\hat{x}) + C \leq f(x') + C \leq f(z) \leq \max_{y \in [\hat{x}, z]} f(y) \).

According to Lemma 3,

\[
f(\hat{x}) = \min_{y \in [\hat{x}, x_t]} f(y)
\]  \hspace{1cm} (5)

From \([x, z] \supset [\hat{x}, z]\) and \( f(x^0) = \min_{y \in [x, z]} f(y) \), \( f(\hat{x}) = \min_{y \in [\hat{x}, z]} f(y) \), we have that \( f(x^0) \leq f(\hat{x}) \). Using \([\hat{x}, z]\) and \( \hat{x} \leq z \), we have that \( f(x^0) \leq \min_{y \in [\hat{x}, x_t]} f(y) \) and \( f(x^0) = \min_{y \in [x, z]} f(y) \leq \min_{y \in [\hat{x}, z]} f(y) \). Finally, since \( I_0 = [x, x_t] = [x, \hat{x}] \cup [\hat{x}, x_t] \), \( D^1 = I_0 \), then \( f(x^0) = \min_{y \in D^1} f(y) \). That is, \( \min_{y \in [x, z]} f(y) = \min_{y \in D^1} f(y) \). \( \square \).

In the sequel we state the analogous lemmas, theorem and corollary for the case where there exists an interval \([a, b]\) in the domain where now \( b \) is the minimum of \( f \). Therefore, all the proofs are omitted since they are completely analogous to previous ones.

**Lemma 4.** If there exist \( x^0 \in D^1 \) and \( b \in \mathbb{Z}^+ \) such that \([x^0 - b, x^0] \subseteq I_0 \) with the following conditions holding

\[
f(x^0) = \min_{y \in [x^0 - b, x^0]} f(y),
\]

\[
f(x^0) + C \leq \max_{y \in [x^0 - b, x^0]} f(y),
\]

then

\[
\min\{w : f(w) = \max_{y \in [x^0 - b, x^0]} f(y)\} \in [x^0 - b, x^0 - \frac{b}{2}] \).
\]

**Lemma 5.** If there exist \( x^0 \in D^1 \) and \( b \in \mathbb{Z}^+ \) such that \([x^0 - b, x^0] \subseteq I_0 \) with the following conditions holding

\[
f(x^0) = \min_{y \in [x^0 - b, x^0]} f(y),
\]

\[
f(x^0) \leq \max_{y \in [x^0 - b, x^0]} f(y)
\]

then \( f(x^0) = \min_{y \in [x, x^0]} f(y) \).

**Theorem 2.** If there exist \( x^0 \in D^1 \) and \( b \in \mathbb{Z}^+ \) such that \([x^0 - b, x^0] \subseteq I_0 \) with the following conditions holding

\[
f(x^0) = \min_{y \in [x^0 - b, x^0]} f(y),
\]

\[
f(x^0) \leq \max_{y \in [x^0 - b, x^0]} f(y)
\]

then \( f(x^0) = \min_{y \in D^1} f(y) \).
Corollary 2. If there exist \( x' \in D^1, z \in [x_s, x'] \) with \( f(z) - f(x') \geq C \), then
\[
\min_{y \in [z, x]} f(y) = \min_{y \in D^1} f(y).
\]

The following theorem is important since it proves that if we can find an interval in which there is a local minimum under some conditions then this is global minimum. To do that we use the properties of the intervals we defined in the above discussion.

Theorem 3. If there exist \( x^0 \in D^1 \) and \( b_1, b_2 \in \mathbb{Z}^+ \), such that \( [x^0 - b_2, x^0 + b_1] \subseteq I_0 \) with the following conditions:
\[
\begin{align*}
    f(x^0) &= \min_{y \in [x^0 - b_2, x^0 + b_1]} f(y), \\
    f(x^0) + C &\leq \max_{y \in [x^0 - b_2, x^0 + b_1]} f(y), \\
    f(x^0) + C &\leq \max_{y \in [x^0 - b_2, x^0]} f(y),
\end{align*}
\]

then \( f(x^0) = \min_{y \in D^1} f(y) \).

Proof: Let \( S_1 = [x^0, x^0 + b_1], S_2 = [x^0 - b_2, x^0] \), then \( f(x^0) = \min_{y \in (S_1 \cup S_2)} f(y), f(x^0) + C \leq \max_{y \in S_1} f(y), f(x^0) + C \leq \max_{y \in S_2} f(y) \).

Let \( I_+ = [x^0, x^1], I_- = [x_s, x^0] \). Since \( f(x^0) = \min_{y \in (S_1 \cup S_2)} f(y) \) and \( x^0 \in S_0, x^0 \in S_1, x^0 \in I_+, x^0 \in I_- \), then
\[ f(x^0) = \min_{y \in S_1} f(y) = \min_{y \in S_2} f(y). \]

Also, because \( f(x^0) = \min_{y \in S_1} f(y), f(x^0) + C \leq \max_{y \in S_1} f(y) \), it follows that \( f(x^0) = \min_{y \in I_+} f(y) \) according to Lemma 3.

Finally, from \( f(x^0) = \min_{y \in S_2} f(y), f(x^0) + C \leq \max_{y \in S_2} f(y) \), it follows that \( f(x^0) = \min_{y \in I_-} f(y) \) according to Lemma 5. Now, since \( I_0 = I_+ \cup I_- \), \( D^1 = I^0 \), we have that \( f(x^0) = \min_{y \in D^1} f(y) \). \( \square \)

Now, we state a corollary which shows how to find the interval with the above property.

Corollary 3. If there exist \( x', z_s, z_t \in D^1 \), such that \( z_s < x', z_t > x' \) with \( f(z_s) - f(x') \geq C \) and \( f(z_t) - f(x') \geq C \)

then
\[
\min_{y \in [z_s, z_t]} f(y) = \min_{y \in D^1} f(y).
\]

Proof: Let \( f(x^0) = \min_{y \in [z_s, z_t]} f(y) \), we need to prove \( \min_{y \in D^1} f(y) = f(x^0) \).

Since, \( f(x^0) = \min_{y \in [z_s, z_t]} f(y) = \min_{y \in [z_s, z_t]} f(y), \) then \( f(x^0) + C \leq f(x') + C \leq f(z_t) \leq \max_{y \in [z_s, z_t]} f(y) \).

Therefore, \( f(x^0) = \min_{y \in [z_s, z_t]} f(y) \) according Lemma 3.

Also, since \( f(x^0) = \min_{y \in [z_s, z_t]} f(y) = \min_{y \in [z_s, z_t]} f(y), \) then \( f(x^0) + C \leq f(x') + C \leq f(z_s) \leq \max_{y \in [z_s, z_t]} f(y) \).

Therefore, \( f(x^0) = \min_{y \in [z_s, z_t]} f(y) \) according Lemma 5.
Finally, $D^1 = I_0 = ([x_s, x^0] \cup [x^0, x_1])$. Hence, $f(x^0) = \min_{y \in D^1} f(y)$. □

Through the above discussion, we have narrowed the computation of the optimal solution of any one-dimension ameso optimization problem into the identification of some intervals given in the above theorems.

The following Example implements the above corollary to minimize a function with domain an ameso set.

**Example 4:** $f(x) = \frac{1}{4}x^4 - x^3 + x$, $x \in [-20, 20]$ is an ameso(4) set from Example 2. At the same time, we have $f(1) - f(3) = 4 \geq 4$, $f(4) - f(3) = 7.75 \geq 4$ and $f(3) = \min\{f(i) : i = 1, 2, 3, 4\}$, then $f(3) = \min\{f(i) : i \in [-20, 20]\}$.

As we mentioned above corollary 3 is useful to obtain an algorithm which can solve an one dimension ameso optimization problem. Therefore, we have the following algorithm.

**one-dimension ameso(C) optimization algorithm**

**Input:** ameso(C) optimization problem:

**minimize** $f(x)$; subject to $x \in D^1$

**Step 1:** $A = \phi$, $l^+ = 0$, $l^- = 0$, select a point $l^0 \in D^1$, $l^* = l^0$, $S = l^0$, then calculate and define as $f^*(l^0) = \min f(x)$,

**Step 2:** Update $l^+ = \min\{x : x > l^0, x \in D^1 - A\}$, then calculate and define as $f_n^*(l^+) = \min f(x)$,

**Step 3:** If $f^*(l^+) - f^*(l^*) \geq C$, go to **step 4**;

- If $0 \leq f^*(l^+) - f^*(l^*) < C$, then $A = A \cup \{l^+\}$ go to **step 2**.
- If $f^*(l^+) - f^*(l^*) < 0$, then $A = A \cup \{l^+\}$, $l^* = l^+$ go to **step 2**.
- If $\{x : x \in D^1 - A, x > l_0\} = \phi$, go to **step 4**;

**Step 4:** Update $l^-$ to be any integer satisfying $f^*(l^-) = \max_{l^* : l^* \in A, l < l^*} f^*(l)$;

- If $f^*(l^-) - f^*(l^*) \geq C$, go to **output**;

**Step 5:** Update $l^- = \max\{x : x < l^0, x \in D^1 - A\}$, then calculate and define as $f^*(l^-) = \min f(x)$

**Step 6:** If $f^*(l^-) - f^*(l^*) \geq C$, go to **output**;

- If $0 \leq f^*(l^-) - f^*(l^*) < C$, then $A = A \cup \{l^-\}$ go to **step 5**.
- If $f^*(l^-) - f^*(l^*) < 0$, then $A = A \cup \{l^-\}$, $l^* = l^-$ go to **step 5**.
- If $\{x : x \in D^1 - A, x < l_0\} = \phi$, go to **output**;

**Output:** $A, l^*, f^*(l^*)$

Below we give an example to show how the algorithm can be implemented.

**Example 5:** Define a function $f(x)$ where the range of $x$ is the set

$\{1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31\}$
with corresponding values

\[ f(x) = 7, 9, 7, 8, 7, 8, 9, 8, 7, 8, 9, 8, 7, 8, 6, 7, 4, 5, 6, 7, 6, 7, 8, 9, 10, 11, 9, 8, 7, \]

i.e., \( f(1) = 7, f(2) = 9, \ldots, f(31) = 8 \). The graph of \( f \) is given figure 1. The problem of minimizing \( f \) over its range is an one-dimensional ameso(7) problem.

![Graph of f](https://via.placeholder.com/150)

Figure 1

If \( l^0 = 13 \), then we can find the global minimum point \( l^* = 17 \) after doing computations involving only the set of points in the set \( \{1, \cdots, 27\} \). Indeed the algorithm will work as follows: first it will search to the right of 13 and it will stop at point \( l^+ = 27 \) because \( f(27) - f(17) = 7 \). The minimum value in the interval \( \{13, \cdots, 27\} \) is \( f(17) = 4 \).

Then the algorithm will compare the maximum value \( f(14) = 8 \) of \( f \) in the interval \( \{13, \cdots, 16\} \) and since \( f(14) - f(17) = 4 < 7 \), it will continue with the following step.

Now, the algorithm will search to the left of 13 and it will stop at point \( l^- = 1 \) because there is no \( x \in \{1, \cdots, 13\} \) satisfying \( f(x) - f(17) \geq 7 \). The algorithm has computed the global minimum point \( l^* = 17 \), without considering points in the set \( \{28, \cdots, 31\} \).

**Remark:** From the definition of an ameso(\( C \)) optimization problem one can see that any discrete optimization problem over a finite set can be transformed into an ameso(\( C \)) optimization problem when \( C \geq 2*(\max\{f(x)\} - \min\{f(x)\}) \), and its domain is an ameso set. However, such a large \( C \) is meaningless, because in order to satisfy the conditions of stopping in the above algorithm we need to check the whole domain if \( C > \max\{f(x)\} - \min\{f(x)\} \). Therefore, it is obvious that an ameso(\( C \)) optimization can be preferred over other discrete optimization methods only if \( C \leq \max\{f(x)\} - \min\{f(x)\} \). In such cases
the above algorithm can narrow down the computations dramatically and as we showed in Example 5. For instance, in the Example 5 the function \( f \) is an ameso(\( C \)) problem, for any \( C \geq 7 \). If we use the algorithm with \( C = 8 \), and starting point at \( l^0 = 13 \), we have going though its steps as above that it will search the whole interval \( \{1, \ldots, 31\} \). This happened since in that case \( C = 8 > 7 = \max\{f(x)\} - \min\{f(x)\} \).

Another question that raises from the above example is which \( C \) to choose since we know that if a problem is ameso(\( C \)) it is also ameso(\( aC \), \( a > 0 \). As we showed in Example 5, the number of computations depends on the starting point and \( C \). Thus, the question is, for the same starting point what \( C \) is preferable?

Now, we provide a corollary which establishes that if a function, defined on an ameso set, is an ameso(\( C_1 \)) and ameso(\( C_2 \)) optimization where \( C_1 < C_2 \), then we prefer to implement the algorithm for the minimum \( C \), i.e., \( C_1 \).

**Corollary 4.** If \( (D^1, f) \) is an ameso(\( C_1 \)) pair then it is an ameso(\( C_2 \)) for any \( C_2 > C_1 \) and \( C_2 \leq \max\{f(x)\} - \min\{f(x)\} \). We need less computations if we apply the algorithm, with the same starting point \( l^0 \), for \( C_1 \).

**Proof:** The proof follows from the following coupling argument. According to the algorithm for an ameso(\( C \)) optimization, we stop the searching (Steps 3 and 6) if we satisfy some inequalities which depend on \( C \). For the upper bound the inequality is \( f^*(l^+ - f^*(l^*) \geq C \) (Step 3) and for the lower bound is \( f^*(l^-) - f^*(l^*) \geq C \) (Step 6). Now, if the ameso(\( C_2 \)) satisfies first the inequalities (first means that \( l^+_{C_1} \leq l^+_{C_2} \) and \( l^-_{C_1} \leq l^-_{C_2} \) so less computations) it is obvious that for some \( l^+_{C_2} \) and \( l^-_{C_2} \), we have that \( f^*(l^+_{C_2}) - f^*(l^*) \geq C_2 \), and \( f^*(l^-_{C_2}) - f^*(l^*) \geq C_2 \) but since \( C_1 < C_2 \) this means that for \( l^+_{C_1} = l^+_{C_2} \) and \( l^-_{C_1} = l^-_{C_2} \) we have that \( f^*(l^+_{C_1}) - f^*(l^*) \geq C_1 \), and \( f^*(l^-_{C_1}) - f^*(l^*) \geq C_1 \). In that case, the number of computations is the same for both \( C_1 \) and \( C_2 \). However, if the ameso(\( C_1 \)) satisfies first the inequalities we have for some \( l^+_{C_1} \leq l^+_{C_2} \) and \( l^-_{C_1} \leq l^-_{C_2} \), that \( f^*(l^+_{C_1}) - f^*(l^*) \geq C_1 \), and \( f^*(l^-_{C_1}) - f^*(l^*) \geq C_1 \) but since \( C_2 > C_1 \) it is not guaranteed that the inequalities hold for \( l^+_{C_1} = l^+_{C_2} \) and \( l^-_{C_1} = l^-_{C_2} \). In that case, the ameso(\( C_2 \)) may need more computations than ameso(\( C_1 \)). Therefore, it is preferred the ameso(\( C_1 \)) based on the number of possible computations when the algorithm uses the same starting point. □

**Remark:** It follows from the above corollary that if our interest is to minimize a discrete function over an ameso domain set with the minimum number of computations then the optimal is to apply the algorithm to the ameso(\( C \)) optimization where this \( C \) is the minimum \( C \) one can obtain.

### 2.3. Properties of multi-dimension ameso(\( C \)) Pair

In this section we introduce the multi-dimensional ameso optimization problem and we discuss its properties. Using these properties we show an algorithm for the multi-dimensional case which is based on the decomposition analysis and the one-dimension ameso optimization problem.

Consider a multi-dimensional ameso(\( C \)) optimization problem: \( \text{minimize } f(\bar{x}); \text{ subject to } \bar{x} \in D^n \). For fixed \( i_1, \ldots, i_j (n \geq j) \) we start with the following definition.
**Definition 4.** 1. The domain $\Delta^j_{i_1, \ldots, i_j}$ is the set:

$$\Delta^j_{i_1, \ldots, i_j} = \{(x_{i_1}, \ldots, x_{i_j}) : \exists (x'_1, \ldots, x'_n) \in D^n \text{ with } x'_{i_k} = x_{i_k}, \forall k = 1, \ldots, j\}.$$ 

2. The conditional domain of $\tilde{x} \in D^n$ given fixed $x^0_{i_1}, \ldots, x^0_{i_j}$ to be the set:

$$\Gamma^{n-j}_{x^0_{i_1}, \ldots, x^0_{i_j}} = \{(x_1, \ldots, x_n) \in D^n : x_{i_k} = x^0_{i_k} \forall k = 1, \ldots, j\}.$$ 

3. The conditional function: $f^*_{i_1, \ldots, i_j} : \Delta^j_{i_1, \ldots, i_j} \rightarrow \mathbb{R}$

$$f^*_{i_1, \ldots, i_j}(x_{i_1}, \ldots, x_{i_j}) = \min_{\tilde{y} \in \Gamma^{n-j}_{x^0_{i_1}, \ldots, x^0_{i_j}}} f(\tilde{y}).$$

4. The conditional pair of ameso(C) pair to be the pair: $(\Delta^j_{i_1, \ldots, i_j}, f^*_{i_1, \ldots, i_j}).$

Now we can establish the next essential property.

**Property 5.** The conditional pair $(\Delta^j_{i_1, \ldots, i_j}, f^*_{i_1, \ldots, i_j})$ of an $n$-dimensional ameso(C) pair $(D^n, f)$ is a $j$-dimensional ameso(C) pair.

**Proof:** At first we prove $\Delta^j_{i_1, \ldots, i_j}$ is an ameso set. And $\forall (x_{i_1}, x_{i_2}, \ldots, x_{i_j}), (y_{i_1}, y_{i_2}, \ldots, y_{i_j}) \in \Delta^j_{i_1, \ldots, i_j}$, we can find a vector $\tilde{x}' = (x'_1, \ldots, x'_n) \in D^n, x'_{i_k} = x_{i_k}, \forall k = 1, \ldots, j$ and a vector $\tilde{y}' = (y'_1, \ldots, y'_n) \in D^n, y'_{i_k} = y_{i_k}, \forall k = 1, \ldots, j$.

And because $D^n$ is an ameso set, so $\tilde{x}', \tilde{y}' \in D^n \Rightarrow \left[\frac{\tilde{x}'+\tilde{y}'}{2}\right], \left[\frac{\tilde{x}'+\tilde{y}'}{2}\right] \in D^n.$

Thus we find $\left[\frac{\tilde{x}'+\tilde{y}'}{2}\right] \in D^n$ with $\left[\frac{x'_{i_k}+y'_{i_k}}{2}\right] = \left[\frac{x_{i_k}+y_{i_k}}{2}\right], \forall k = 1, \ldots, j$ and

$$\left[\frac{\tilde{x}'+\tilde{y}'}{2}\right] \in D^n$$ \quad with \quad $\left[\frac{x'_{i_k}+y'_{i_k}}{2}\right] = \left[\frac{x_{i_k}+y_{i_k}}{2}\right], \forall k = 1, \ldots, j.$

That is to say,

$$(\left[\frac{x'_{i_1}+y'_{i_1}}{2}\right], \ldots, \left[\frac{x'_{i_j}+y'_{i_j}}{2}\right]), (\left[\frac{x_{i_1}+y_{i_1}}{2}\right], \ldots, \left[\frac{x_{i_j}+y_{i_j}}{2}\right]) \in \Delta^j_{i_1, \ldots, i_j}$$ \quad (6)

So $\Delta^j_{i_1, \ldots, i_j}$ is an ameso set.

Next we show the second condition of the definition. For any $(x_{i_1}, x_{i_2}, \ldots, x_{i_j}), (y_{i_1}, y_{i_2}, \ldots, y_{i_j}) \in \Delta^j_{i_1, \ldots, i_j}$, $
\exists \tilde{x}_0 \in D^n, \tilde{y}_0 \in D^n, \quad f(\tilde{x}_0) = f^*_{i_1, \ldots, i_j}(x_{i_1}, x_{i_2}, \ldots, x_{i_j}), f(\tilde{y}_0) = f^*_{i_1, \ldots, i_j}(y_{i_1}, y_{i_2}, \ldots, y_{i_j}).$ Since $(D^n, f)$ is an ameso(C) pair, we have

$$f^*_{i_1, \ldots, i_j}(x_{i_1}, x_{i_2}, \ldots, x_{i_j}) + f^*_{i_1, \ldots, i_j}(y_{i_1}, y_{i_2}, \ldots, y_{i_j}) + C \geq f(\tilde{x}_0) + f(\tilde{y}_0) + C \geq f\left(\left[\frac{\tilde{x}_0+\tilde{y}_0}{2}\right]\right)$$

$$\geq f^*_{i_1, \ldots, i_j}\left(\left[\frac{x_{i_1}+y_{i_1}}{2}\right], \ldots, \left[\frac{x_{i_j}+y_{i_j}}{2}\right]\right) + f^*_{i_1, \ldots, i_j}\left(\left[\frac{x_{i_1}+y_{i_1}}{2}\right], \ldots, \left[\frac{x_{i_j}+y_{i_j}}{2}\right]\right)$$

(because of and the definition of $f^*_{i_1, \ldots, i_j}()$).
That is to say \((\Delta_{i_1,\ldots,i_j}^l, f_{i_1,\ldots,i_j}^*)\) is also an ameso\((C)\) pair, and \(f_{i_1,\ldots,i_j}^*\) is a \(j\)-dim function \(\square\).

Now, using the above property we can construct an algorithm which obtains the minimum of a multi-dimension ameso optimization problem. Therefore, we establish the following theorem.

**Theorem 4.** The solution of any ameso optimization problem can be found using the Ameso Reduction Procedure(ARP), described in the table below.

| Ameso Reduction Procedure (ARP) |
|----------------------------------|
| **Input:** | ameso\((C)\) Optimization problem: minimize \(f(\mathbf{x})\); subject to \(\mathbf{x} \in \mathbb{D}^n\) |
| **Step 1:** | \(A = \phi, l^+ = 0, l^- = 0\), select a point \(l^0 \in \Delta^1_n, l^* = l^0, S = l^0\), then calculate \(f_n^*(l^0) = \min_{(x_1,\ldots,x_{n-1},l^0) \in \Gamma_{l^0}^{n-1}} f(x) \uparrow\) |
| **Step 2:** | Update \(l^+ = \min\{x : x > l^0, x \in \Delta^1_n - A\}\), then calculate \(f_n^*(l^+) = \min_{(x_1,\ldots,x_{n-1},l^+) \in \Gamma_{l^+}^{n-1}} f(x) \uparrow\) |
| **Step 3:** | If \(f_n^*(l^+) - f_n^*(l^*) \geq C\), go to **Step 4**; |
| & | If \(0 \leq f_n^*(l^+) - f_n^*(l^*) < C\), then \(A = A \cup \{l^+\}\) go to **Step 2**. |
| & | If \(f_n^*(l^+) - f_n^*(l^*) < 0\), then \(A = A \cup \{l^+\}, l^* = l^+\) go to **Step 2**. |
| & | If \(\{x : x \in \Delta^1_n - A, x > l_0\}\) \(\phi\), go to **Step 4**; |
| **Step 4:** | Update \(l^-\) to be any integer satisfying \(f_n^*(l^-) = \max_{l : l \in A, l < l^*} f_n^*(l)\); |
| & | If \(f_n^*(l^-) - f_n^*(l^*) \geq C\), go to **output**; |
| **Step 5:** | Update \(l^- = \max\{x : x < l^0, x \in \Delta^1_n - A\}\), then calculate \(f_n^*(l^-) = \min_{(x_1,\ldots,x_{n-1},l^-) \in \Gamma_{l^-}^{n-1}} f(x) \uparrow\) |
| **Step 6:** | If \(f_n^*(l^-) - f_n^*(l^*) \geq C\), go to **output**; |
| & | If \(0 \leq f_n^*(l^-) - f_n^*(l^*) < C\), then \(A = A \cup \{l^-\}\) go to **Step 5**. |
| & | If \(f_n^*(l^-) - f_n^*(l^*) < 0\), then \(A = A \cup \{l^-\}, l^* = l^-\) go to **Step 5**. |
| & | If \(\{x : x \in \Delta^1_n - A, x < l_0\}\) \(\phi\), go to **output**; |
| **Output:** | \(A, l^*, f_n^*(l^*)\) |

\(\uparrow\)Note: the computation of \(f_n^*(l^0), f_n^*(l^+)\) and \(f_n^*(l^-)\) above involves solution \((n - 1)\)-dimension ameso\((C)\) optimization problems.

**Proof:** The proof is easy to complete using Property 5 and Theorem 3. \(\square\)

Below, we give an example with a 2-dimension ameso optimization problem and we implement the ARP algorithm we provided above.

**Example 6:** Consider the function \(f(x_1, x_2) = 88e^{x_1^2} + 99e^{x_2^2} + \frac{\sin(x_1x_2)}{2}\), and the domain \(D^2 = \{(x_1, x_2) : x_i = 1, 2, \cdots, 100, i = 1, 2\}\). The problem of minimizing \(f(x)\) over \(D^2\) is a two-dimension ameso\((1)\) optimization.

Assume we pick \(x_2 = 80\) as the starting point. Then \(l^0 = 80\). The ARP will calculate \(f_2^*(80) =\)
\[ \min_{(x_1, 80) \in \Gamma_{80}^1} f(x_1, 80) = 190.4220, \text{ where } \Gamma_{80}^1 = \{(x_1, x_2) : x_1 = 1, \ldots, 100; x_2 = 80\}. \] Then it will begin to search in \( x_2 \)'s increasing side, i.e., \( l^+ = 81, 82, 83, \ldots \). It will keep on calculating \( f_2^*(l^+) = \min_{(x_1, l^+) \in \Gamma_{l^+}^1} f(x_1, l^+) \). In the end it will stop at \( l^+ = 100 \), and the the minimum point in the interval \( x_2 \in \{80, \ldots, 100\} \) is \( f_2^*(97) = \min_{(x_1, 97) \in \Gamma_{97}^1} f(x_1, 97) = f(97, 97) = 190.0093 \).

Next we calculate the difference of the maximum point in the set \( \{80, \ldots, 96\} \) and \( f_2^*(97) \), that is: \( f_2^*(81) - f_2^*(97) < 1 = C \). Hence the ARP will next search in the direction of the decreasing side of \( x_2(x_2 < 80) \), it will compute \( f_2^*(l^-) \) for \( l^- = 79, 78, \ldots \) and it stop when \( l^- = 66 \) with \( f_2^*(66) - f_2^*(97) = 191.1640 - 190.0093 \geq 1 \).

Now we can say that \( x_1^* = 97, x_2^* = 97 \) is a global minimum with value \( f(x_1^*, x_2^*) = 190.0093 \).

We illustrate the function \( f \) and the ARP technique with the following figures. In figure 2 we graph the function for all points in its domain \( D^2 \). In figure 3 we graph the function for a smaller part of its domain. In figure 4 we plot the function only for points that were involved in our ARP technique. Note that the number of points in figure 4 is significantly smaller than the points in Figure 3 and figure 2. In figure 5 we plot the function \( f_2^*(x_2) \), that is calculated in our ARP technique. It is an one-dimension ameso(1) function.

![Figure 2](image-url)
Figure 3

Figure 4
Remark: In Example 6, if we choose any point \( x_2 \) greater than 65 as starting point, then the \( \text{ARP} \) will stop after a search of all \( l^+ \in \{x_2 + 1, \ldots, 100\} \) and \( l^- \in \{x_2 - 1, \ldots, 66\} \). If choose as starting point \( x_2 \) one that is smaller than 66, then the \( \text{ARP} \) will stop after a search of all \( l^+ \in \{x_2 + 1, \ldots, 100\} \), and it will not do a search in the left side of \( x_2 \).

Remark: From Examples 5 and 6, we can see that the choice of starting points and the form of the functions themselves determine the complexity of \( \text{ARP} \). For an arbitrary integral optimization problem, if one can establish that it is an ameso(\( C \)) optimization for a suitable number \( C \), then one can use the \( \text{ARP} \) to find the optimal solution without necessarily searching all points in the domain.

3. Conclusion

In this paper we introduced the class of Ameso(\( C \)) optimization problems. We also established that the main properties they have are:

1. for one-dimensional ameso optimization problems there are simple, to verify, optimality conditions at any optimal point (see Theorems 1, 2, 3 and corresponding corollaries),
2. the conditional pair of ameso(\( C \)) pair is still an ameso(\( C \)) pair (see Property 5),
3. we have constructed an \( \text{ARP} \) that solves multi-dimension ameso optimization problems without necessarily performing complete enumeration. (see \( \text{ARP} \)).
Further, we recall, that one can prove that a problem is ameso($C$) optimization in the same way we prove convexity or midpoint convexity. Therefore, we can see that an ameso($C$) optimization is a relaxed convex model, since in convex cases we have ameso(0) optimization problem.

As we mentioned in Section 2.2, there is a difficulty to satisfy that the constant $C$ is less than the difference of the maximum and minimum of the corresponding function we study. In many problems this difference is given so one can compare it with the constant $C$ that comes from the ameso optimization model. However, there are models where this difference can not be provided by direct model analysis. For cases where the difference of the maximum and minimum is not known and there is no way to estimate it, in order to be sure that we will not enumerate the whole domain, one can try to prove that the problem is is an ameso(1) optimization problem. Since for any non constant function the difference of maximum and minimum is grater than 1 we are sure that an ameso($C$) with $C = 1$ satisfies the condition. Thus, for models with functions where it is not possible to prove convexity or midpoint convexity it may still be possible to show that the more relaxed property of the ameso(1) optimization holds. To do that is easier than to prove convexity and at the same time there is the ARP algorithm that can compute the minimum point without requiring complete enumeration.

Acknowledgments We acknowledge support for this work from the National Science Foundation, NSF grant CMMI-16-62629.

References

References

[1] Bertsekas, D. P. and Tsitsiklis, J. N. (2003). “Parallel and Distributed Computation: Numerical Methods”, LIDS Technical Reports, MIT, Boston Ma.

[2] Bertsimas, D.J. and van Ryzin, G.J. (1993), “Stochastic and Dynamic Vehicle Routing with General Interarrival and Service Time Distributions”. Advances in Applied Probability, 25, pp. 947-78

[3] Bertsimas, D.J. and van Ryzin, G.J. (1990), “An Asymptotic Determination of the Minimum Spanning Tree and Minimum Matching Constants in Geometrical Probability,” Operations Research Letters, 9, pp. 223-231.

[4] De Loera, J.A., Hemmecke, R., Koppe, M. (2013), “Algebraic and Geometric Ideas in the Theory of Discrete Optimization.” SIAM, Philadelphia.

[5] Denardo, E. V., Huberman, G. and Rothblum, U. G. (1982) “Optimal Locations on a Line Are Interleaved”, Operations Research, 30, pp. 745-759.

[6] Hemmecke, R., Koppe, M., Lee, J., Weismantel, R. (2010), “Nonlinear integer programming.” In: Junger, M., et al. (eds.) 50 Years of Integer Programming 1958-2008: From The Early Years and State-of-the-Art, Chapter 15, pp. 561-618, Springer, Berlin.
[7] Gittins, J., Glazebrook, K., and Weber, R. (2011). “Multi-armed bandit allocation indices”, John Wiley & Sons.

[8] Katehakis, M. N., and Veinott Jr, A. F. (1987). “The multi-armed bandit problem: decomposition and computation”, Mathematics of Operations Research, 12(2), 262-268.

[9] Lee, J., Leyffer, S. (2012), “Mixed Integer Nonlinear Programming.” The IMA Volumes in Mathematics and its Applications 154, Springer, Berlin.

[10] Lewis, H. R and Papadimitriou, C. H (1998) “Elements of the Theory of Computation”, ACM Press, New York.

[11] Moriguchi, S., Murota, K., Tamura, A., and Tardella, F. (2017). “Discrete Midpoint Convexity.” https://arxiv.org/pdf/1708.04579.pdf.

[12] Onn, S. (2010), “Nonlinear Discrete Optimization: An Algorithmic Theory.” European Mathematical Society, Zurich.

[13] Papadimitriou, C. H. and Steiglitz, K. (1982). “Combinatorial Optimization: Algorithms and Complexity”, Publisher: Prentice-Hall.

[14] Papadimitriou, C. H. and Yannakakis, M. (1982) “The complexity of facets (and some facets of complexity)” Annual ACM Symposium on Theory of Computing archive Proceedings of the fourteenth annual ACM symposium on Theory of computing, San Francisco, California, United States, pp. 255 - 260.

[15] Topkis, D. M. (1978). “Minimizing a Submodular Function on a Lattice”, Operations Research, 26, pp. 305-321.

[16] Yannakakis, M. (1982)”The complexity of the partial order dimension problem”, SIAM J. Alg. Dis. Meth.,3, pp. 351-370.

[17] Yannakakis, M. (1990). “The analysis of local search problems modified greedy heuristic for the set covering problem and their heuristics”, STAG’S -90, pp. 298-311.

[18] Zipkin P., (2016) “Some specially structured assemble-to-order systems.” Operations Research Letters, 44, pp. 136-142.