CompAMa
www.compama.co.usb.ve

*Bull. Comput. Appl. Math., Vol.4, No.2, 2016*
ISSN 2244-8659

# Modified spectral projected subgradient method: convergence analysis and momentum parameter heuristics

*Milagros Loreto* [*], *Samantha Clapp* [†], *Charles Cratty* [‡], *Breeanna Page* [§]

CompAMa Vol.4, No.2, pp.27-54, 2016 - Accepted September 17, 2016
In Honor of the 60th Birthday of Professor Marcos Raydan

## Abstract

The Modified Spectral Projected Subgradient ($MSPS$) was proposed to solve Langrangen Dual Problems, and its convergence was shown when the momentum term was zero. The $MSPS$ uses a momentum term in order to speed up its convergence. The momentum term is built on the multiplication of a momentum parameter and the direction of the previous iterate. In this work, we show convergence when the momentum parameter is a non-zero constant. We also propose heuristics to choose the momentum parameter intended to avoid the Zigzagging Phenomenon of Kind I. This phenomenon is present in the $MSPS$ when at an iterate the subgradient forms an obtuse angle with the previous direction. We identify and diminish the Zigzagging Phenomenon of Kind I on Setcovering problems, and compare our numerical results to those of the original $MSPS$ algorithm.

[*]School of STEM, University of Washington Bothell, Bothell, WA, 98011, USA (mloreto@uwb.edu).

[†]Department of Mathematics, Georgia College and State University, Milledgeville, GA, 31061, USA (samantha.clapp@bobcats.gcsu.edu).

[‡]Department of Math and Computer Science, Westminster College, New Wilmington, Pa, 16142, USA (cratcd22@wclive.westminster.edu).

[§]Department of Mathematics, Eastern Washington University, Cheney, WA, 99004, USA (breeannapage@eagles.ewu.edu).

**Keywords:** Spectral projected gradient, Subgradient methods, Momentum term.

## 1 Introduction

The $MSPS$ introduced by Loreto et al. [1] evolves from the spectral projected subgradient ($SPS$)[2], incorporating ideas from the $SPG1$ method proposed by Raydan et al. [3, 4] to calculate the direction on each iteration. $MSPS$ was proposed to solve Lagrangean dual problems with non-differentiable convex objective function, and it presents theoretical grounds as its convergence was shown when the momentum parameter was zero. The $SPS$ is based on the $SPG2$ method proposed by Raydan et al. [3, 4], and although it does not present convergence grounds, it has shown good performance, presenting several advantages such as: step length independent of the optimal value; acceleration approaches like the momentum term; low memory requirements; and low computation cost per iteration. $MSPS$ inherits all of these advantages, while presenting stronger convergence properties. Indeed, the search direction for the $MSPS$ is the negative direction of the subgradient and a projection is computed for each trial point, which allows to obtain the Euclidean distance to the optimal set, supporting the convergence proof.

The momentum term is a convergence acceleration technique that was adapted to the $MSPS$ and the $SPS$. This technique helps to avoid the typical zigzagging behavior of the subgradient methods and also accelerates their convergence. The momentum term was adopted from the work proposed by Plaut et al. [5], where it was used as an acceleration technique to perform numerical experimentation with Back Propagation learning algorithms in neural networks. In [6], the Zigzagging Phenomenon of Kind I is defined as a motivation to use this parameter, although it was named as deflection parameter.

For the non-monotone globalization technique, the $MSPS$ combines and extends the Grippo et al. [7] line search scheme with the proposed globalization scheme of La Cruz et al. [8]. Also, for theoretical convergence the spectral step length sequence is bounded by a sequence that converges to zero, and is non-summable.

In this work, our first target is to develop a comprehensive convergence analysis for the $MSPS$ when the momentum parameter is constant. Our convergence proof is based on the Euclidean distance to the optimal value

and the non-monotone globalization condition. Our second target is to develop a heuristic to dynamically calculate the momentum parameter at each iteration. The proposed heuristic is designed to overcome the Zigzagging Phenomenon of Kind I, which occurs in the $MSPS$ when at an iterate $\lambda_k$, the previous direction $m_k$ and the subgradient $g_k$ form an obtuse angle. We present a geometric perspective of this heuristic. Hence, the $MSPSd\tau_k$ is presented as a new version of the $MSPS$ incorporating this heuristic.

This paper is organized as follows: In Section 2, we present the $MSPS$ method and background concepts such as: spectral step, globalization condition, and momentum term. In Section 3, we present our convergence analysis for a constant momentum parameter. In Section 4, we develop heuristics to calculate the momentum parameter based on our geometrical analysis. In Section 5, we present numerical results on set covering problems, and compare the performance of the $MSPSd\tau_k$ against the original $MSPS$. In Section 6, we present our final remarks.

## 2 $MSPS$ Review

In this section we briefly describe the $MSPS$ as it was presented in [1]. This method was proposed to solve the problem (D):

$$\begin{array}{ll} \min & f(\lambda) \\ \text{s.t.} & \lambda \in \Omega, \end{array}$$

where $f(\lambda) = \max\{c^T x + \lambda^T(b - Ax), x \in X\}$ is a convex and piecewise linear function, and it is non-differentiable at some points, $\Omega = \{\lambda : \lambda \geq 0\}$, $X$ is a finite set, $x, c \in \mathbb{R}^n, \lambda \in \mathbb{R}^m$, and $A \in \mathbb{R}^{m \times n}$.

To get an optimal point $\lambda_*$ that minimizes $f(\lambda)$, the $MSPS$ computes a trial point $\lambda_+$ in the negative direction of $m_+$, which is the trial direction based on the subgradient $g_k = g(\lambda_k)$, the spectral step $\alpha_k$ , and the momentum real parameter $\tau$. The following procedure is used to obtain $m_+$ and $\lambda_+$, with $m_0 = 0$.

**Procedure (A):**

1. $m_+ = \alpha_k g_k + \tau m_k$
2. $\lambda_+ = P_\Omega(\lambda_k - m_+)$, where $P_\Omega(\lambda)$ is defined as the projection of $\lambda$ on $\Omega$

$P_\Omega(\lambda) = \lambda_+$ and its components are defined by $(\lambda_+)_i = \max\{0, \lambda_i\}$, for $i = 1, \ldots, m$. The point $\lambda_+$ is tested until it satisfies the globalization condition (3) and a new iterate is obtained. The structure of this iteration is very attractive because of its simplicity. The convergence of the $MSPS$ was proved under some mild assumptions when the momentum parameter was zero. All the elements part of the $MSPS$ are explained in subsections 2.1, 2.2, and 2.3.

## 2.1 Spectral Steplength $\alpha_k$

Let us briefly describe the spectral step, that will be used for the $MSPS$ to solve the problem D. In the differentiable case, the basic idea to obtain the spectral step length is to regard the matrix $(1/\alpha_k)I$ as an approximation to the Hessian $\nabla^2 f(\lambda_k)$ and impose a quasi-Newton secant equation

$$\frac{1}{\alpha_k} s_{k-1} = y_{k-1},$$

where $s_{k-1} = \lambda_k - \lambda_{k-1}$ and $y_{k-1} = \nabla f(\lambda_k) - \nabla f(\lambda_{k-1})$. In general, this equation cannot be solved. However, accepting the least-squares solution that minimizes $\|\frac{1}{\alpha_k} s_{k-1} - y_{k-1}\|_2^2$, we obtain the so-called spectral step length:

$$\alpha_k = \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}.$$

By the Mean-Value Theorem of integral calculus, it follows that:

$$y_{k-1} = \left( \int_0^1 \nabla^2 f(\lambda_{k-1} + t s_{k-1}) dt \right) s_{k-1}.$$

Hence, $\alpha_k$ is the inverse of a Rayleigh quotient relative to the average Hessian matrix

$$\int_0^1 \nabla^2 f(\lambda_{k-1} + t s_{k-1}) dt,$$

and so it is between the minimum and the maximum eigenvalue of the average Hessian, which has been the main motivation for using the word *spectral* in its name. Additional properties and variations of the spectral step length can be found in [9, 10].

Writing the secant equation as $(1/\alpha_k)y_{k-1} = s_{k-1}$, which is also standard in the quasi-Newton tradition, we arrive at a different spectral coefficient:

$$\alpha_k = \frac{s_{k-1}^T y_{k-1}}{y_{k-1}^T y_{k-1}}.$$

Now the formulas for $\alpha_k$ are used, provided that for all iterations they are bounded away from zero on the positive side and bounded away from infinity. In other words, we use some safeguard fixed parameters $0 < \alpha_{\min} < \alpha_{\max} < \infty$ and define at each iteration:

$$\alpha_k = \min\{\alpha_{\max}, \max\{\alpha_{\min}, \frac{s_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}}\}\}, \tag{1}$$

which will be our spectral choice of step length throughout the rest of this work.

Since the convergence analysis of subgradient schemes is based on the fact that the sequence $\{\| \lambda_{k+1} - \lambda_* \|\}$ is strictly decreasing, the following condition is added to the spectral step $\alpha_k$, as it is proposed by Polyak in [11].

$$\alpha_k > 0 \quad \forall k, \quad \lim_{k\to\infty} \alpha_k = 0, \quad \sum_{k=1}^{\infty} \alpha_k = \infty. \tag{2}$$

To guarantee condition (2), we bound the spectral step by the sequence $\{\frac{C}{\log(k)}\}$ with $C > 0$ which satisfies (2). Here is a procedure to verify this condition :

**Procedure (B):**

1. if $\alpha_{k+1} \geq \frac{10^8}{\log(k)}$ then $\alpha_{k+1} = \frac{10^8}{\log(k)}$
2. if $\alpha_{k+1} \leq \frac{10^{-8}}{\log(k)}$ then $\alpha_{k+1} = \frac{10^{-8}}{\log(k)}$

## 2.2 Non-monotone Globalization Technique

For our non-monotone globalization technique, we combine and extend the Grippo et al. [7] line search scheme with the proposed globalization scheme

of La Cruz et al. [8]. Roughly speaking, our acceptance condition (i.e. globalization condition) for the next iterate is:

$$f(\lambda_{k+1}) \leq \max_{0\leq j\leq \min\{k,M\}} f(\lambda_{k-j}) + \gamma\rho_k(\lambda_{k+1}-\lambda_k)^t g_k + \eta_k \tag{3}$$

where $\gamma$ is a small positive number, the vector $g_k$ is the gradient or subgradient, $\rho_k > 0$ is obtained after a backtracking process that starts at 1, and $\eta_k$ is chosen such that:

$$0 < \sum_{k=0}^{\infty} \eta_k < \infty. \tag{4}$$

The terms $\max_{0\leq j\leq M} f(\lambda_{k-j})$ and $\eta_k > 0$ are responsible for the sufficiently nonmonotone behavior of $f(\lambda_k)$. In practice, the parameters associated with the line search strategy are chosen to reduce the number of backtracking instances as much as possible, while keeping the convergence properties of the method. For example, the parameter $\gamma > 0$ is chosen as a very small number such as $10^{-4}$ and $\eta_k$ for $k = 0$ is chosen as a large number $\eta_0 = \max(f(\lambda_0), \| g(\lambda_0) \|)$, and then is reduced as slow as possible, making sure that (4) holds.

## 2.3 Momentum Term

A typical iteration using the momentum term is proposed in [5], where it is used as an acceleration technique to perform numerical experimentation with Back Propagation learning algorithms, and it is widely studied in [12]. The problem of learning in neural networks is formulated in terms of the minimization of the error function $E$. This error is a function of weight parameters $w_k$ and $\nabla E(w_k) = v_k$. The use of the momentum term can be described as follows:

$$w_{k+1} = w_k - \Delta v_k,$$
$$\Delta v_k = \sigma_k v_k + \tau \Delta v_{k-1},$$

where $\tau\Delta v_{k-1}$ is the momentum term, and $\tau \in [0, 1]$ is the momentum parameter, $\sigma_k$ is the learning rate, $w_0$ and $\Delta v_0$ are given. This technique is studied in more detail in section 4.

## 2.4 *MSPS* Algorithm

Given $\lambda_0 \in \Omega$, a parameter *maxiter* representing the maximum number of iterations allowed, $\alpha_0 > 0$, $\tau \in \mathbb{R}$, and $\gamma = 10^{-4}$. Set $k = 0$ and $m_0 = 0$.

- Compute $f(\lambda_0)$, $g(\lambda_0)$, and set $\eta_0 = \max(f(\lambda_0), \| g(\lambda_0) \|)$.
- While $k <$ *maxiter*
  1. Obtain $m_+$ and $\lambda_+$ by procedure (A) with spectral step $\alpha_k$ and $\tau$ constant
  2. Set $\rho_k = \alpha_k$ and $\eta_k = \frac{\eta_0}{k^{1.1}}$
  3. Backtracking: While condition (3) is not satisfied
     Reduce $\rho_k$
     Update $m_+$ and $\lambda_+$ by procedure (A) using $\rho_k$ instead of $\alpha_k$
  4. Set $\lambda_{k+1} = \lambda_+$ and $m_{k+1} = m_+$
  5. Set $s_k = \lambda_{k+1} - \lambda_k$ and $y_k = g_{k+1} - g_k$
  6. Compute $\alpha_{k+1}$ using equation (1) and verify Procedure (B)
  7. Update $g_k$, $f_k$, and set $k = k + 1$
- End While.

## 3 Convergence Analysis

In this section, we present a convergence analysis for $MSPS$ when the momentum parameter $\tau$ is set to a constant value $0 < \tau < 1$. We base the convergence analysis on the non-monotone globalization technique, [7,8]. For standard projected gradient descent methods, the convergence proof is based on the function value decreasing at each iterate. However, for methods such as the $MSPS$, which is a projected subgradient-type method, the proof is based on the Euclidean distance to the optimal value instead. In order to develop an expression to compute such distance, we first characterize a typical iteration of the $MSPS$ for a constant $\tau$ through the following lemma.

**Lemma 1.** *Let $\{\lambda_k\} \subseteq \Omega$ be a sequence of iterates generated by the $MSPS$ applied to the problem $D$, $\lambda_*$ the optimal solution, and $\tau$ a fixed number in $(0, 1)$. Suppose that $\Omega$ is a convex subset of $\mathbb{R}^n$. Then:*

$$\| \lambda_{k+1} - \lambda_* \|_2^2 \leq \| \lambda_k - \lambda_* \|_2^2 - 2 \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j (\lambda_k - \lambda_*)^T g_j \right) + \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j \|g_j\|_2 \right)^2 .$$

*Proof.* Let $\lambda_{k+1} = P_\Omega(z_{k+1})$ be a standard $MSPS$ update, where $m_{k+1} = \alpha_k g_k + \tau m_k$, $\tau \in (0,1)$ and constant. Notice that: $\lambda_0 > 0$, $\alpha_0 > 0$ , $m_0 = 0$, and $g_0 = g(\lambda_0)$.

Given $z_{k+1} = \lambda_k - m_{k+1}$, applying recursion on $m_{k+1}$, we have:

$$\begin{aligned}
z_{k+1} &= \lambda_k - m_{k+1} \;=\; \lambda_k - \alpha_k g_k - \tau m_k \\
&= \lambda_k - \alpha_k g_k - \tau(\alpha_{k-1} g_{k-1} + \tau m_{k-1}) \\
&= \lambda_k - \alpha_k g_k - \tau(\alpha_{k-1} g_{k-1} - \tau(\alpha_{k-2} g_{k-2} + \tau m_{k-2})) \\
&= \lambda_k - \alpha_k g_k - \tau\alpha_{k-1} g_{k-1} - \tau^2 \alpha_{k-2} g_{k-2} - \tau^3 m_{k-2} \\
&= \lambda_k - \alpha_k g_k - \tau\alpha_{k-1} g_{k-1} - \cdots - \tau^{k-1}\alpha_1 g_1 - \tau^k \alpha_0 g_0 - \tau^{k+1} m_0 \\
&= \lambda_k - \alpha_k g_k - \tau\alpha_{k-1} g_{k-1} - \cdots - \tau^{k-1}\alpha_1 g_1 - \tau^k \alpha_0 g_0 \\
&= \lambda_k - \sum_{j=0}^{k} \tau^{k-j} \alpha_j g_j. \qquad (5)
\end{aligned}$$

Consider the distance between the iterate $\lambda_{k+1}$ and the optimal solution $\lambda_*$. It is true that:

$$\|\lambda_{k+1} - \lambda_*\|_2 = \|P_\Omega(z_{k+1}) - \lambda_*\|_2 \leq \|z_{k+1} - \lambda_*\|_2.$$

Because of (5),

$$\|\lambda_{k+1} - \lambda_*\|_2 \leq \|z_{k+1} - \lambda_*\|_2 = \|\lambda_k - \sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j - \lambda_*\|_2.$$

Hence,

$$\begin{aligned}
\|\lambda_{k+1} - \lambda_*\|_2^2 \leq \|\lambda_k - \sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j - \lambda_*\|_2^2 \;&=\; \|\,(\lambda_k - \lambda_*) - \sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j\|_2^2 \\
= \|\lambda_k - \lambda_*\|_2^2 - 2(\lambda_k - \lambda_*)^T \left(\sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j\right) + \|\sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j\|_2^2 \\
= \|\lambda_k - \lambda_*\|_2^2 - 2\left(\sum_{j=0}^{k} \tau^{k-j}\alpha_j (\lambda_k - \lambda_*)^T g_j\right) + \|\sum_{j=0}^{k} \tau^{k-j}\alpha_j g_j\|_2^2
\end{aligned}$$

$$\leq \|\lambda_k - \lambda_*\|_2^2 - 2\left(\sum_{j=0}^{k} \tau^{k-j}\alpha_j(\lambda_k - \lambda_*)^T g_j\right) + \left(\sum_{j=0}^{k} |\tau^{k-j}||\alpha_j|\|g_j\|_2\right)^2$$

$$= \|\lambda_k - \lambda_*\|_2^2 - 2\left(\sum_{j=0}^{k} \tau^{k-j}\alpha_j(\lambda_k - \lambda_*)^T g_j\right) + \left(\sum_{j=0}^{k} \tau^{k-j}\alpha_j\|g_j\|_2\right)^2$$

□

Since the $MSPS$ is not a descent method, we keep track of the best point found so far, (*i.e.*, the one with smallest function value).
At each step, we set:

$$f_k^{best} = \min\{f_{k-1}^{best}, f(\lambda_k)\},\ k = 1, 2, \ldots, \text{with}\ \ f_0^{best} = f(\lambda_0).$$

If $\lambda_k$ is the best point found, we have:

$$f_k^{best} = \min\{f(\lambda_0), f(\lambda_1), \ldots, f(\lambda_{k-1}), f(\lambda_k)\},$$

where $(f_k^{best})$ is the best objective value found in $k$ iterations. As $f_k^{best}$ is a non-increasing sequence, it has a limit.

Since the sequence $\alpha_k$ generated by $MSPS$ algorithm satisfies the condition (2) and $\tau$ is a constant number less than one, the sequence $\{\lambda_k\}$ generated by the $MSPS$ is guaranteed to converge to the optimal value. It means $\lim_{k\to\infty} f_k^{best} = f_*$, where $f_*$ denotes the optimal value of the problem.

**Theorem 1.** *Let $\{\lambda_k\} \subseteq \Omega$ be a sequence of iterates generated by the $MSPS$ applied to the problem $D$, with spectral step $\alpha_k$ satisfying the condition (2). Let the momentum parameter $\tau$ be a constant number in $(0, 1)$, and suppose an optimal solution $\lambda_*$ exists. Suppose the $MSPS$ is applied with the additional assumption that there exists $G > 0$, such that $\| g \|_2 \leq G$ for all $g \in \partial f(\lambda)$, where $\partial f(\lambda)$ is the set of subgradient vectors of $f(\lambda)$. Then:*

$$\lim_{k\to\infty} f_k^{best} = f_*.$$

*Proof.* Based on lemma (1), we have:

$$\begin{aligned}
&\| \lambda_{k+1} - \lambda_* \|_2^2 \\
&\leq \| \lambda_k - \lambda* \|_2^2 - 2 \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j (\lambda_k - \lambda_*)^T g_j \right) + \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j \|g_j\|_2 \right)^2 \\
&\leq \| \lambda_k - \lambda* \|_2^2 - 2 \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j (\lambda_k - \lambda_*)^T \sum_{j=0}^{k} g_j \right) + \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j \|g_j\|_2 \right)^2 .
\end{aligned} \tag{6}$$

To find a bound for $-2 \left( \sum_{j=0}^{k} \tau^{k-j} \alpha_j (\lambda_k - \lambda_*)^T \sum_{j=0}^{k} g_j \right)$, we use the globalization condition (3). Therefore, if $\lambda_*$ is an optimal solution then the globalization condition is satisfied for any subgradient $g_k$ at $\lambda_k$:

$$f_* \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(\lambda_{k-j}) + \gamma (\lambda_* - \lambda_k)^T g_k + \eta_k$$

and it is also true for $\sum_{j=0}^{k} g_j$, which is a subgradient at $\lambda_k$. Consequently, the equation above is equivalent to:

$$f_* \leq \max_{0 \leq j \leq \min\{k, M-1\}} f(\lambda_{k-j}) + \gamma (\lambda_* - \lambda_k)^T (\sum_{j=0}^{k} g_j) + \eta_k \implies$$

$$\max_{0 \leq j \leq \min\{k, M-1\}} f(\lambda_{k-j}) - f_* + \eta_k \geq \gamma (\lambda_k - \lambda_*)^T (\sum_{j=0}^{k} g_j) \implies$$

$$(\lambda_k - \lambda_*)^T (\sum_{j=0}^{k} g_j) \leq \frac{1}{\gamma} (\max_{0 \leq j \leq \min\{k, M-1\}} f(\lambda_{k-j}) - f_* + \eta_k). \tag{7}$$

Combining (6) and (7), and using that $\| g \|_2 \leq G$ for all $g \in \partial f(\lambda)$, we get:

$$\begin{aligned}
&\| \lambda_{k+1} - \lambda_* \|_2^2 \leq \| \lambda_k - \lambda* \|_2^2 - \\
&\frac{2}{\gamma} (\sum_{j=0}^{k} \tau^{k-j} \alpha_j) (\max_{0 \leq j \leq \min\{k, M-1\}} f(\lambda_{k-j}) - f_* + \eta_k) + (\sum_{j=0}^{k} \tau^{k-j} \alpha_j)^2 G^2.
\end{aligned} \tag{8}$$

Applying (8) recursively:

$$\begin{aligned}
&\| \lambda_{k+1} - \lambda_* \|_2^2 \leq \| \lambda_0 - \lambda* \|_2^2 \\
&- \frac{2}{\gamma}(\sum_{j=0}^{0} \tau^{0-j}\alpha_j)(\max_{0\leq j\leq \min\{0,M-1\}} f(\lambda_{0-j}) - f_* + \eta_0) + (\sum_{j=0}^{0} \tau^{0-j}\alpha_j)^2 G^2 \\
&+ \cdots + (-\frac{2}{\gamma}(\sum_{j=0}^{k-1} \tau^{k-1-j}\alpha_j)(\max_{0\leq j\leq \min\{k-1,M-1\}} f(\lambda_{k-1-j}) - f_* + \eta_{k-1}) + \\
&(\sum_{j=0}^{k-1} \tau^{k-1-j}\alpha_j)^2 G^2) + (-\frac{2}{\gamma}(\sum_{j=0}^{k} \tau^{k-j}\alpha_j)(\max_{0\leq j\leq \min\{k,M-1\}} f(\lambda_{k-j}) - f_* + \eta_k) \\
&+ (\sum_{j=0}^{k} \tau^{k-j}\alpha_j)^2 G^2).
\end{aligned}$$

Grouping like terms:

$$\begin{aligned}
\| \lambda_{k+1} - \lambda_* \|_2^2 \leq &\| \lambda_0 - \lambda* \|_2^2 \\
&- \frac{2}{\gamma}(\sum_{j=0}^{0} \tau^{0-j}\alpha_j)(\max_{0\leq j\leq \min\{0,M-1\}} f(\lambda_{0-j}) - f_* + \eta_0) + \dots \\
&+ (-\frac{2}{\gamma}(\sum_{j=0}^{k-1} \tau^{k-1-j}\alpha_j)(\max_{0\leq j\leq \min\{k-1,M-1\}} f(\lambda_{k-1-j}) - f_* + \eta_{k-1}) \\
&+ (-\frac{2}{\gamma}(\sum_{j=0}^{k} \tau^{k-j}\alpha_j)(\max_{0\leq j\leq \min\{k,M-1\}} f(\lambda_{k-j}) - f_* + \eta_k) \\
&+ G^2((\sum_{j=0}^{0} \tau^{0-j}\alpha_j)^2 + \dots + (\sum_{j=0}^{k-1} \tau^{k-1-j}\alpha_j)^2 + (\sum_{j=0}^{k} \tau^{k-j}\alpha_j)^2)
\end{aligned}$$

which is equivalent to:

$$\begin{aligned}
\| \lambda_{k+1} - \lambda_* \|_2^2 \leq &\| \lambda_0 - \lambda* \|_2^2 \\
&- \frac{2}{\gamma}\left(\sum_{i=0}^{k}(\sum_{j=0}^{i} \tau^{k-j}\alpha_j)(\max_{0\leq j\leq \min\{i,M-1\}} f(\lambda_{i-j}) - f_* + \eta_i)\right) \\
&+ G^2\left(\sum_{i=0}^{k}(\sum_{j=0}^{i} \tau^{k-j}\alpha_j)^2\right)
\end{aligned} \tag{9}$$

$$
\begin{aligned}
=&\| \lambda_0 - \lambda* \|_2^2 - \frac{2}{\gamma}\left(\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)(\max_{0\le j\le \min\{i,M-1\}} f(\lambda_{i-j}) - f_*)\right) \\
&- \frac{2}{\gamma}\left(\sum_{i=0}^{k}\eta_i(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)\right) + G^2\left(\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)^2\right).
\end{aligned}
$$

Since $\| \lambda_{k+1} - \lambda_* \|_2^2 \geq 0$, and $\| \lambda_0 - \lambda* \|_2^2$ is a constant number $R$, we have:

$$
\begin{aligned}
0 \leq & R - \frac{2}{\gamma}\left(\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)(\max_{0\le j\le \min\{i,M-1\}} f(\lambda_{i-j}) - f_*)\right) + \\
& - \frac{2}{\gamma}\left(\sum_{i=0}^{k}\eta_i(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)\right) + G^2\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)^2.
\end{aligned} \tag{10}
$$

The term $\max_{0\le j\le \min\{i,M-1\}}(f(\lambda_{i-j}) - f_*)$ is bounded below by $(f_k^{best} - f_*)$ as it was shown in [1, pp. 924-925], therefore eq.(10) is equivalent to:

$$
\begin{aligned}
&\frac{2}{\gamma}(\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j))(f_k^{best} - f_*) \\
&\qquad \leq R - \frac{2}{\gamma}(\sum_{i=0}^{k}\eta_i(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)) + G^2\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)^2.
\end{aligned} \tag{11}
$$

Solving (11) for $\left(f_k^{best} - f(\lambda_*)\right)$, we arrive at the most general basic inequality:

$$
\begin{aligned}
(f_k^{best} - f(\lambda_*)) \leq & \frac{\gamma R}{2\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)} - \frac{\sum_{i=0}^{k}\eta_i(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)}{\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)} \\
& + \frac{\gamma G^2\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)^2}{2\sum_{i=0}^{k}(\sum_{j=0}^{i}\tau^{k-j}\alpha_j)}.
\end{aligned} \tag{12}
$$

For Cauchy Product definition,

$$
\sum_{i=0}^{k}\left(\sum_{j=0}^{i}\tau^{k-j}\alpha_j\right) = \left(\sum_{i=0}^{k}\alpha_i\right)\left(\sum_{j=0}^{k}\tau^j\right),
$$

and substituting it in (12), the inequality (12) becomes:

$$f_k^{best} - f(\lambda_*) \leq \left( \frac{\gamma R}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right) - \left( \frac{\left(\sum_{i=0}^{k} \eta_i\right)\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)}{\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right) + \left( \frac{\gamma G^2 \left(\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)\right)^2}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right). \tag{13}$$

Hence, the inequality 13 becomes:

$$f_k^{best} - f(\lambda_*) \leq \left( \frac{\gamma R}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right) - \left( \sum_{i=0}^{k} \eta_i \right) + \left( \frac{\gamma G^2 \left(\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)\right)^2}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right). \tag{14}$$

The geometric series $\sum_{j=0}^{k} \tau^j$ converges since $\tau < 1$ and $\sum_{i=0}^{k} \eta_i$ is a convergence series by definition, hence the convergence of the right hand side of (14) is driven by the series $\sum_{i=0}^{k} \alpha_i$.

In (14), the sequence: $\left( \frac{\gamma R}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right)$ converges to zero as $k \to \infty$, since the denominator grows without a bound, and the geometric series $\sum_{j=0}^{k} \tau^j$ converges for $\tau < 1$.

Also, in (14) the convergence of sequence: $\left( \frac{G^2 \left(\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)\right)^2}{2\left(\sum_{i=0}^{k} \alpha_i\right)\left(\sum_{j=0}^{k} \tau^j\right)} \right)$ is driven by the convergence of the sequence $\frac{\left(\sum_{i=0}^{k} \alpha_i\right)^2}{\left(\sum_{i=0}^{k} \alpha_i\right)}$.

By the square sum rule, we have:

$$\frac{\left(\sum_{i=0}^{k}\alpha_i\right)^2}{\left(\sum_{i=0}^{k}\alpha_i\right)} = \frac{\left(\sum_{i=0}^{k}{\alpha_i}^2 + 2\sum_{i<k}\alpha_i\alpha_k\right)}{\left(\sum_{i=0}^{k}\alpha_i\right)} = \frac{\sum_{i=0}^{k}{\alpha_i}^2}{\sum_{i=0}^{k}\alpha_i} + \frac{2\sum_{i<k}\alpha_i\alpha_k}{\sum_{i=0}^{k}\alpha_i}. \quad (15)$$

Under the assumption of the condition(2), $\frac{\sum_{i=0}^{k}{\alpha_i}^2}{\sum_{i=0}^{k}\alpha_i}$ converges as it was shown in [1, pp. 926]. Also, $\frac{\sum_{i<k}\alpha_i\alpha_k}{\sum_{i=0}^{k}\alpha_i}$ converges because the numerator converges and the denominator grows without a limit. Indeed, $\sum_{i<k}\alpha_i\alpha_k$ converges since $\sum_i \alpha_i$ is bounded by a convergent sequence (Procedure B), and $\lim_{k\to\infty}\alpha_k = 0$.

Finally, since all the sequences on the right hand side of (14) converge then the $\lim_{k\to\infty} f_k^{best} = f_*$ □

# 4 Momentum Parameter Heuristics

In this section, we describe two heuristics for choosing the momentum parameter $\tau$ used by the $MSPS$.

The momentum term is a remedy for the slowness of the steepest descent. It is built on the multiplication of $\tau$ and the previous direction, and it is added to the current gradient. This term helps to avoid the typical zigzagging behavior of the steepest descent method and also accelerates its convergence. The same zigzagging behavior is present in subgradient and projected subgradient methods; therefore, it was adapted to the $SPS$ in [2] and $MSPS$ in [1].

## 4.1 Constant-$\tau$ Heuristic

This heuristic was originally proposed in [2] and used in [1], where the momentum parameter $\tau$ is set to a constant value less than one. It is worth to mention that the line search used in this heuristic reduces only the portion of $m_k$ that contains $\alpha_k$, while the momentum term is kept unchanged, as can be seen in the step 3 of the $MSPS$ algorithm described in section 2.4. In both works, the numerical experimentation showed that it was clever to keep

the momentum term unaffected by the line search backtracking when $\tau$ is a constant, and $\tau = 0.7$ was the recommended value yielding the best results. As shown in the previous convergence analysis, this heuristic is guaranteed to converge.

## 4.2 Dynamic-$\tau_k$ Heuristic

The momentum parameter, also called deflected parameter by some authors, has been used in works such as [13] by Tseng and [6] by Guta et al., among others. In particular, the Zigzagging Phenomenon of Kind I is defined in [6] as a motivation to use this deflected parameter.

**Zigzagging of Kind I:** Let $m_k$ a direction vector and $g_k$ a subgradient vector. The iterative procedure:

$$\lambda_{k+1} = P_\Omega(\lambda_k - m_k), \quad k = 0, 1, 2 \ldots$$

forms a *Zigzagging of Kind I* if at any two (or more) consecutive iterate points, the angle between corresponding directions $m_k$ and $g_k$ is obtuse; i.e., $m_k^T g_k < 0$.

Zigzagging of Kind I occurs in the $MSPS$ when at an iterate $\lambda_k$, the previous direction $m_k$ and the subgradient $g_k$ form an obtuse angle $\beta$. This phenomenon is represented in Figure (1).

Based on the geometrical interpretation of the momentum term, we propose the Dynamic-$\tau_k$ Heuristic to choose $\tau_k$ on each iteration in order to accelerate the convergence and to avoid Zigzagging of Kind I. Notice that $\tau_k$ will replace $\tau$ in Procedure A.

As a means to develop the heuristic to calculate $\tau_k$, we denote the angle between $m_k$ and $g_k$ as $\beta$, the angle between $m_+$ and $m_k$ as $\theta$, and the angle between $m_+$ and $g_k$ as $\omega$. These labeling of the angles can be seen in Figure (2). Notice $m_+ = \alpha_k g_k + \tau_k m_k$ as it is defined in the $MSPS$, and clearly $\theta + \omega = \beta$.

To avoid the zigzagging we aim to deduce $\tau_k$ in such way that $\theta$ becomes an acute angle. As it is seen in Figure (3), the Zigzagging of Kind I will not

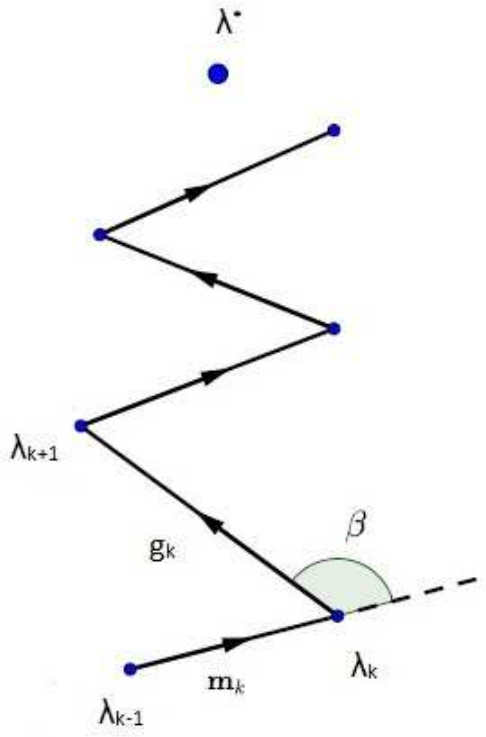


Fig. 1: Zigzagging

happen if $\theta$ is acute. Therefore, we derive the $\tau_k$-Formula, which is an upper bound for $\tau_k$ based on a desirable $\theta$, by using the dot product between $m_+$ and $m_k$. From now on, we define the norm two as $\| \, . \, \|_2 = \| \, . \, \|$.

**Proposition 1.** *Let $m_+$ and $m_k$ be defined as in the MSPS and $\theta$ the angle between $m_+$ and $m_k$. An upper bound for $\tau_k$ is defined as:*

$$\tau_k \leq \frac{\alpha_k}{1 - \frac{1}{\cos\theta}} \left[ \frac{g_k^T m_k}{\|m_k\|^2 \cos\theta} - \frac{\|g_k\|}{\|m_k\|} \right]$$

*for $\theta \neq \dfrac{\pi}{2}$. This upper bound is called $\tau_k$-Formula.*

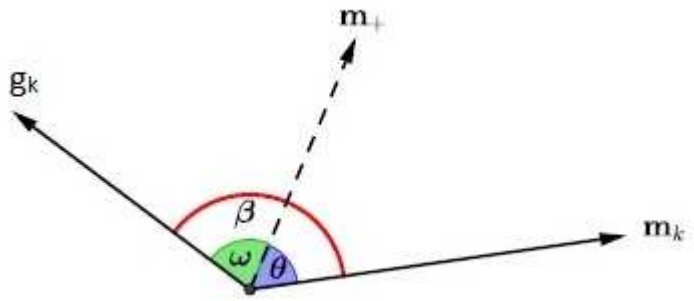


Fig. 2: Labeling Angles

*Proof.*

$$
\begin{aligned}
m_+^T m_k &= \|m_+\|\|m_k\| \cos\theta \\
\left(\alpha_k g_k + \tau_k m_k\right)^T m_k &= \|\alpha_k g_k + \tau_k m_k\|\|m_k\| \cos\theta \\
\frac{\alpha_k g_k^T m_k + \tau_k \|m_k\|^2}{\|m_k\| \cos\theta} &= \|\alpha_k g_k + \tau_k m_k\| \leq \alpha_k \|g_k\| + \tau_k \|m_k\| \\
\frac{\tau_k \|m_k\|^2}{\|m_k\| \cos\theta} - \tau_k \|m_k\| &\leq \alpha_k \|g_k\| - \frac{\alpha_k g_k^T m_k}{\|m_k\| \cos\theta} \\
\tau_k \|m_k\| \left[\frac{1}{\cos\theta} - 1\right] &\leq \alpha_k \left[\|g_k\| - \frac{g_k^T m_k}{\|m_k\| \cos\theta}\right] \\
\tau_k &\leq \frac{\alpha_k}{1 - \frac{1}{\cos\theta}} \left[\frac{g_k^T m_k}{\|m_k\|^2 \cos\theta} - \frac{\|g_k\|}{\|m_k\|}\right].
\end{aligned}
$$

□

### 4.2.1 Geometrical Interpretation

In this subsection, we analyze both cases, when $\beta$ is obtuse and acute, and we describe the motivation behind the Dynamic-$\tau_k$ heuristic.

**Case One:** $\beta > \dfrac{\pi}{2}$.
In this case, the main goal is to choose $\tau_k$ in such way that the angle $\theta$ between the resulting direction $m_+$ and the previous direction $m_k$ is acute, i.e.

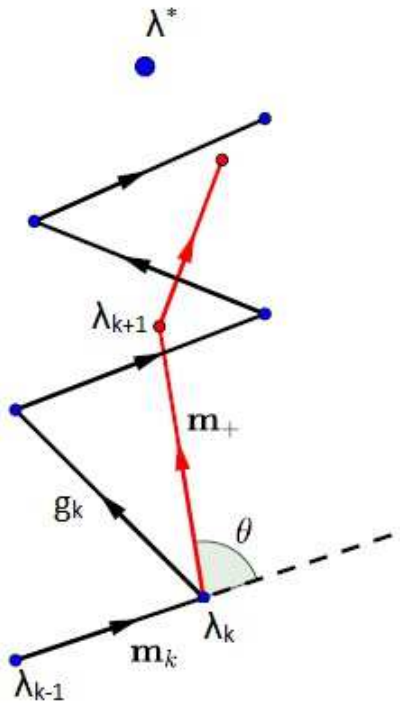


Fig. 3: Acute $\theta$

$\theta < \frac{\pi}{2}$, a condition we named as the $\theta$-condition.

The heuristic chooses the minimum $\tau_k$ over the set $T = \{t \in \mathbb{R} \mid t = r10^{-1},\ r \in \mathbb{N},\ r \in [1, 10]\}$ that satisfies the $\theta$-condition. Figure (4) illustrates this situation; the red line represents the boundary where zigzagging is avoided since $\theta$ is acute. The dashed lines represent possible choices for $m_+$ based on different values of $\tau_k$. The selected direction $m_+$ would be the one closest to the boundary in the red region.

In some cases, none of the candidates for $\tau_k$ in the set T satisfies the $\theta$-condition. This is illustrated in Figure (5), where the dashed lines representing prospective $m_+$ do not fall within the acceptable region. For this

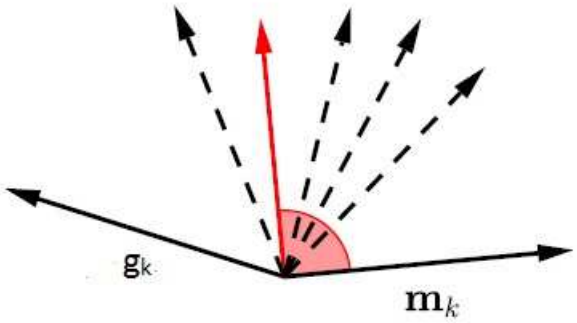


Fig. 4: $\theta$-Bound

case, we use the $\tau_k$-Formula to calculate an upper bound of $\tau_k$ where $\theta = \frac{\beta}{2}$ is the target angle. In other words, we try to obtain an approximation of $\tau_k$ to bisect $\beta$.

However, since the $\tau_k$-Formula represents an upper bound of $\tau_k$, it could produce values much greater than one for some iterations; in such case we bound $\tau_k$ to two as shown in (16). The rationale behind bounding $\tau_k$ to two is that, although we want to increase the weight of the previous direction over $m_+$ to avoid zigzagging, we don't want to disregard the information provided by $g_k$.

$$\tau_k = \min\{\frac{\alpha_k}{1 - \frac{1}{\cos(\frac{\beta}{2})}} \left[ \frac{g_k^T m_k}{\|m_k\|^2 \cos(\frac{\beta}{2})} - \frac{\|g_k\|}{\|m_k\|} \right], 2\} \tag{16}$$

**Case Two:** $\beta \leq \frac{\pi}{2}$ .
In this case, even though the momentum parameter could safely be set to zero without any risk of Zigzagging of Kind I, our numerical experimentation revealed that choosing a momentum parameter $\tau_k$ in such way that $\theta \approx \frac{\beta}{2}$ produces better numerical results. Thus we use $\tau_k$-Formula to calculate an upper bound of $\tau_k$ using $\theta = \frac{\beta}{2}$ as our target angle. If the calculated $\tau_k$ is

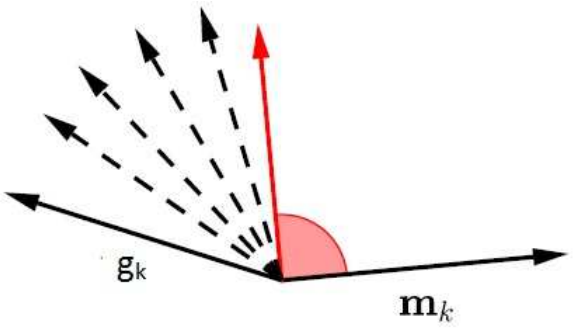


Fig. 5: $\theta$-Bound and $\tau_k$ Fails

greater than one, we bound it to one as shown in (17).

$$\tau_k = \min\{\frac{\alpha_k}{1 - \frac{1}{\cos\left(\frac{\beta}{2}\right)}} \left[\frac{g_k^T m_k}{\|m_k\|^2 \cos\left(\frac{\beta}{2}\right)} - \frac{\|g_k\|}{\|m_k\|}\right], 1\} \tag{17}$$

Based on the angel's interpretation developed above, we now present the heuristic to compute $\tau_k$.

**Dynamic-$\tau_k$ Procedure:**
Given $m_k$, $g_k$, and $\alpha_k$.

1. Compute $\beta$
2. If $\beta > \dfrac{\pi}{2}$

   Find the minimum $\tau \in T$ such that $\theta$-condition is satisfied

   If none of the $\tau$'s satisfies the $\theta$-condition, compute $\tau$ using equation (16)
3. Else $\beta \leq \dfrac{\pi}{2}$

   Compute $\tau$ using equation (17)
4. Set $\tau_k = \tau$

## 4.3 $MSPSd\tau_k$

In this section we present a new version of the $MSPS$ called $MSPSd\tau_k$, which includes the heuristic Dynamic-$\tau_k$ to choose a suitable momentum parameter, and also modifies the backtracking implementation from the original $MSPS$.

$MSPSd\tau_k$ **Algorithm:**

Given $\lambda_0 \in \Omega$, a parameter *maxiter* representing the maximum number of iterations allowed, $\alpha_0 > 0$, and $\gamma = 10^{-4}$. Set $m_0 = 0$ and $k = 0$.

- Compute $f(\lambda_0)$, $g(\lambda_0)$, and set $\eta_0 = \max(f(\lambda_0), \| g(\lambda_0) \|)$.
- While $k <$ *maxiter*
  1. Obtain $\tau_k$ using Dynamic-$\tau_k$ Procedure
  2. Obtain $m_+$ and $\lambda_+$ by procedure (A) with spectral step $\alpha_k$ and momentum parameter $\tau_k$
  3. Set $\rho = 1$ and $\eta_k = \frac{\eta_0}{k^{1.1}}$
  4. Backtracking: While condition (3) is not satisfied
     reduce $\rho$
     Update $\lambda_+ = P_\Omega(\lambda_k - \rho * m_+)$, where $P_\Omega(\lambda)$ is the projection of $\lambda$ on $\Omega$
  5. Set $\lambda_{k+1} = \lambda_+$ and $m_{k+1} = m_+$
  6. Set $s_k = \lambda_{k+1} - \lambda_k$ and $y_k = g_{k+1} - g_k$
  7. Compute $\alpha_{k+1}$ using equation (1) and verify Procedure (B)
  8. Update $g_k$, $f_k$, and set $k = k + 1$
- End While.

The backtracking for the $MSPSd\tau_k$ was implemented in the classical way. Indeed, the direction $m_+$ is computed before the backtracking starts and through successive reductions a final $\lambda_+$ is accepted without computing a new $m_+$. The backtracking in the original $MSPS$ only reduces the portion of $m_+$ that contains $\alpha_k$, and a new $m_+$ is computed. The numerical experimentation showed that it was clever for the $MSPS$ to keep the momentum term unaffected in the backtracking when $\tau$ is constant. However, when using a dynamic $\tau_k$, the classical backtracking yields better results.

# 5 Numerical Results

We compare the performance of $MSPSd\tau_k$ and $MSPS$ on set covering test problems. These algorithms were implemented using Matlab 2013. The experiments were executed on a laptop equipped with Intel Core i5 2.60 GHz, 16 GB of RAM, and running Windows 7 64-bits.

## 5.1 Set Covering Problems

A set covering problem is the problem of covering the rows of an $m$-row, $n$-column, zero-one matrix $(a_{ij})$ by a subset of the columns at minimum cost. These problems can be formulated as follows:

$$\begin{array}{ll} \min & \sum_{j=1}^{n} c_j x_j \\ \text{s.t.} & \\ & \sum_{j=1}^{n} a_{ij} x_j \geq 1 \quad i = 1, \ldots, m \\ & x_j \in \{0,1\} \end{array}$$

with $c_j > 0$, $a_{ij} \in \{0,1\}$, $\forall i, j$.

If we define $\widehat{c_j} = -c_j$ and $\widehat{a_{ij}} = -a_{ij}$, then the problem $P_\lambda$ can be written as

$$\begin{array}{ll} \max & \sum_{j=1}^{n} \widehat{c_j} x_j + \sum_{i=1}^{m} \lambda_i (-1 - \sum_{j=1}^{n} \widehat{a_{ij}} x_j) \\ \text{s.t.} & x_j \in \{0,1\}. \end{array}$$

An optimal solution of $P_\lambda$ is given by

$$x_j = \begin{cases} 1 & \text{if } (\widehat{c_j} - \sum_{i=1}^{m} \lambda_i \widehat{a_{ij}}) > 0 \\ \{0,1\} & \text{if } (\widehat{c_j} - \sum_{i=1}^{m} \lambda_i \widehat{a_{ij}}) = 0 \\ 0 & \text{if } (\widehat{c_j} - \sum_{i=1}^{m} \lambda_i \widehat{a_{ij}}) < 0. \end{cases}$$

Let $x^*$ be a solution of $P_\lambda$ then the subgradient is given by

$$g_i = -1 - \sum_{j=1}^{n} \widehat{a_{ij}} x_j^* \quad i = 1, \ldots, m$$

Tab. 1: Characteristics of set covering problems

| Problem | Rows ($m$) | Columns ($n$) | Density (%) |
|---|---|---|---|
| 4 | 200 | 1000 | 2 |
| 5 | 200 | 2000 | 2 |
| 6 | 200 | 1000 | 5 |
| A | 300 | 3000 | 2 |
| B | 300 | 3000 | 5 |
| C | 400 | 4000 | 2 |
| D | 400 | 4000 | 5 |

In Table 1, we describe the set of covering problems considered in our experiments. All of them were obtained from Beasley's OR library [14]. The parameter *density* represents the percentage of nonzero elements.

Concerning the initial guess, we follow the ideas developed in [15], that have proven to be effective for set covering problems: Set $\lambda_i = \min_{j \in J_i} c_j/|I_j|$ where the set of indices are defined as $J_i = \{j : a_{ij} = 1\}$ and $I_j = \{i : a_{ij} = 1\}$.

The parameter $M$ was set to $M = 10$ for both $MSPS$ and $MSPSd\tau_k$, which is the standard value recommended in the literature [16]. For the $MSPS$, $\tau$ was fixed to $\tau = 0.7$.

For these set of problems, we report *itbest*, the iteration where the best value is obtained before reaching 500 (*maxiter*) iterations. The best value obtained by the $MSPSd\tau_k$ algorithm is reported under the column ($MSPS\tau_k^*$), the best values obtained by the $MSPS$ algorithm are reported under the columns ($MSPS^*$), and $f_*$ represents the optimal solution for the problems. We also report *zig* as the number of iterations where zigzagging was not corrected by the momentum term, *time* is the execution time in seconds, and *Obt* is the number of iterations where the angle $\beta$ was obtuse.

Table 2 shows the impact of the momentum term reducing the Zigzagging of Kind I. The iterations with zigzagging, reported as $zig$, for both $MSPSd\tau_k$ and $MSPS$ with $\tau = 0.7$ are drastically decreased when compared to $MSPS$ with $\tau = 0$. Regarding the number of iterations, the $MSPS$ with $\tau = 0.7$ trends to find the its best solution sooner than $MSPSd\tau_k$, but its approximations to the solutions are not better than those found by $MSPSd\tau_k$. Indeed, fig.(6) shows the quality of the solutions based on the relative error, and it

Tab. 2: Results for set covering problems

| Probs. | $MSPSd\tau_k$ | | | | | $MSPS/\tau = 0.7$ | | | | | $MSPS/\tau = 0$ | | | | $f_*$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $MPS\tau_k^*$ | *itbest* | *zig* | *time* | *Obt* | $MSPS^*$ | *itbest* | *zig* | *time* | *Obt* | $MSPS^*$ | *itbest* | *zig* | *time* | |
| 4.1 | -428.9973 | 211 | 0 | 10.27 | 1 | -428.9945 | 251 | 0 | 14.74 | 1 | -387.9195 | 500 | 1 | 8.58 | -429.0000 |
| 4.2 | -512.0000 | 184 | 0 | 9.39 | 1 | -512.0000 | 125 | 0 | 9.69 | 2 | -435.9432 | 500 | 6 | 8.80 | -512.0000 |
| 4.3 | -516.0000 | 301 | 0 | 9.46 | 3 | -516.0000 | 285 | 0 | 8.95 | 2 | -407.0152 | 500 | 6 | 8.62 | -516.0000 |
| 4.4 | -493.8309 | 495 | 0 | 8.89 | 6 | -493.7344 | 498 | 0 | 8.88 | 2 | -411.5008 | 500 | 4 | 8.79 | -494.0000 |
| 4.5 | -512.0000 | 149 | 0 | 8.77 | 2 | -512.0000 | 123 | 0 | 8.60 | 2 | -435.4363 | 500 | 4 | 8.57 | -512.0000 |
| 4.6 | -556.9829 | 500 | 0 | 9.07 | 1 | -557.0892 | 499 | 0 | 9.02 | 2 | -497.1565 | 500 | 3 | 8.88 | -557.2500 |
| 4.8 | -429.9968 | 354 | 0 | 8.80 | 3 | -429.9997 | 364 | 0 | 9.36 | 2 | -361.6493 | 500 | 4 | 8.61 | -430.0000 |
| 4.2 | -488.3201 | 498 | 0 | 8.89 | 1 | -488.0203 | 498 | 0 | 8.88 | 2 | -419.3402 | 500 | 3 | 8.85 | -488.6666 |
| 4.9 | -638.3747 | 495 | 0 | 8.65 | 1 | -637.9334 | 498 | 0 | 8.72 | 2 | -554.4532 | 500 | 5 | 8.55 | -638.0000 |
| 4.10 | -513.4967 | 218 | 0 | 12.52 | 1 | -513.4066 | 142 | 0 | 21.25 | 1 | -458.1796 | 500 | 1 | 8.72 | -514.0000 |
| 5.1 | -251.1457 | 498 | 0 | 15.72 | 3 | -251.0685 | 500 | 0 | 15.73 | 2 | -209.5783 | 500 | 127 | 15.68 | -251.2250 |
| 5.2 | -299.6417 | 500 | 0 | 15.70 | 4 | -299.5397 | 500 | 0 | 15.70 | 3 | -250.3537 | 500 | 6 | 15.62 | -299.7611 |
| 5.3 | -225.9765 | 188 | 0 | 22.06 | 3 | -225.9811 | 145 | 0 | 28.68 | 2 | -205.1099 | 500 | 6 | 15.61 | -226.0000 |
| 5.4 | -240.4962 | 500 | 0 | 15.68 | 1 | -240.4673 | 494 | 0 | 15.66 | 2 | -208.5410 | 500 | 3 | 15.62 | -240.5000 |
| 5.5 | -210.9983 | 217 | 0 | 22.14 | 1 | -210.9977 | 115 | 0 | 45.34 | 2 | -183.2728 | 500 | 33 | 15.64 | -211.0000 |
| 5.6 | -212.4831 | 221 | 0 | 18.76 | 1 | -212.4706 | 202 | 0 | 25.97 | 2 | -192.9788 | 500 | 10 | 15.63 | -212.0000 |
| 5.7 | -291.5537 | 500 | 0 | 15.70 | 3 | -291.4292 | 490 | 0 | 15.77 | 2 | -255.1685 | 500 | 4 | 15.68 | -291.7778 |
| 5.8 | -286.7357 | 499 | 0 | 15.66 | 3 | -286.7306 | 500 | 0 | 15.66 | 2 | -238.8473 | 500 | 47 | 15.57 | -287.0000 |
| 5.9 | -278.9977 | 500 | 0 | 15.77 | 3 | -278.7652 | 496 | 0 | 15.86 | 1 | -248.9780 | 500 | 2 | 15.54 | -279.0000 |
| 5.10 | -264.9824 | 146 | 0 | 18.71 | 1 | -264.9584 | 61 | 0 | 15.93 | 2 | -214.1272 | 500 | 41 | 15.61 | -265.0000 |
| 6.1 | -133.0928 | 500 | 0 | 11.05 | 4 | -132.8803 | 500 | 0 | 11.06 | 2 | -88.2279 | 500 | 169 | 11.02 | -133.1396 |
| 6.2 | -140.3474 | 500 | 0 | 11.56 | 7 | -140.0836 | 498 | 0 | 11.63 | 2 | -100.4732 | 499 | 414 | 11.55 | -140.4565 |
| 6.3 | -139.6963 | 500 | 0 | 11.03 | 1 | -139.6582 | 497 | 0 | 11.06 | 2 | -97.8171 | 500 | 186 | 11.00 | -140.1340 |
| 6.4 | -128.8809 | 475 | 0 | 11.16 | 3 | -128.8811 | 461 | 0 | 11.23 | 3 | -89.6189 | 500 | 4 | 11.08 | -129.0000 |
| 6.5 | -152.8811 | 499 | 0 | 11.03 | 3 | -152.7412 | 498 | 0 | 11.18 | 3 | -106.2737 | 500 | 292 | 11.00 | -153.3529 |
| A.1 | -246.3698 | 500 | 0 | 30.63 | 1 | -246.0465 | 498 | 0 | 30.72 | 2 | -195.5550 | 500 | 9 | 30.61 | -246.8368 |
| A.2 | -247.1537 | 500 | 0 | 30.74 | 3 | -246.7837 | 498 | 0 | 30.77 | 2 | -185.2496 | 500 | 51 | 30.57 | -247.4964 |
| A.3 | -227.7353 | 500 | 0 | 30.58 | 1 | -227.7051 | 500 | 0 | 30.65 | 2 | -175.7327 | 500 | 17 | 30.50 | -228.0000 |
| A.4 | -231.0685 | 492 | 0 | 30.71 | 1 | -230.7480 | 496 | 0 | 30.71 | 2 | -175.2468 | 500 | 139 | 30.58 | -231.3968 |
| A.5 | -234.8292 | 500 | 0 | 30.76 | 3 | -234.3683 | 500 | 0 | 30.80 | 2 | -181.0990 | 500 | 49 | 30.72 | -234.8889 |
| B.1 | -64.3317 | 486 | 0 | 45.35 | 4 | -64.0930 | 490 | 2 | 45.95 | 4 | -44.5323 | 500 | 250 | 45.07 | -64.5417 |
| B.2 | -69.1840 | 498 | 0 | 45.40 | 9 | -68.8771 | 499 | 4 | 46.11 | 8 | -45.1949 | 500 | 372 | 45.08 | -69.3019 |
| B.3 | -74.0566 | 497 | 0 | 45.20 | 12 | -73.8111 | 497 | 0 | 45.34 | 2 | -45.9073 | 500 | 451 | 45.07 | -74.1572 |
| B.4 | -71.1706 | 499 | 0 | 45.33 | 6 | -70.8106 | 493 | 3 | 46.28 | 16 | -47.0324 | 500 | 364 | 45.14 | -71.2160 |
| B.5 | -67.6103 | 500 | 0 | 45.13 | 2 | -67.3565 | 499 | 3 | 46.58 | 10 | -42.1970 | 500 | 465 | 45.03 | -67.6661 |
| C.1 | -223.4869 | 497 | 0 | 51.78 | 2 | -223.3028 | 500 | 0 | 51.99 | 2 | -155.4842 | 500 | 126 | 51.69 | -223.8010 |
| C.2 | -212.4706 | 500 | 0 | 51.68 | 5 | -211.9659 | 500 | 0 | 51.94 | 2 | -148.8794 | 500 | 379 | 51.62 | -212.8475 |
| C.3 | -234.3147 | 500 | 0 | 51.78 | 2 | -233.7853 | 500 | 0 | 52.04 | 2 | -167.0874 | 500 | 90 | 51.71 | -234.5829 |
| C.4 | -213.4426 | 499 | 0 | 51.81 | 3 | -213.0100 | 500 | 0 | 51.98 | 2 | -162.1379 | 500 | 170 | 51.72 | -213.8483 |
| C.5 | -211.2982 | 499 | 0 | 51.85 | 2 | -211.1757 | 499 | 0 | 52.06 | 2 | -148.8203 | 500 | 103 | 51.78 | -211.6365 |
| D.1 | -55.1238 | 498 | 0 | 84.91 | 4 | -54.7041 | 500 | 1 | 86.24 | 7 | -31.2199 | 500 | 461 | 84.51 | -55.3088 |
| D.2 | -59.2097 | 500 | 0 | 84.71 | 7 | -58.8940 | 495 | 4 | 86.41 | 7 | -35.0277 | 500 | 444 | 84.53 | -59.3454 |
| D.3 | -64.8752 | 493 | 0 | 85.23 | 17 | -64.6898 | 497 | 5 | 86.91 | 12 | -35.3635 | 500 | 489 | 84.46 | -65.0666 |
| D.4 | -55.6749 | 496 | 0 | 85.00 | 3 | -55.3953 | 498 | 1 | 85.80 | 5 | -31.8547 | 500 | 483 | 84.59 | -55.8415 |
| D.5 | -58.5301 | 496 | 0 | 85.28 | 4 | -58.0493 | 500 | 2 | 86.84 | 8 | -32.3035 | 500 | 466 | 84.80 | -58.6155 |
| E.1 | -3.4432 | 469 | 77 | 8.97 | 405 | -3.4412 | 477 | 178 | 8.43 | 347 | -3.2745 | 500 | 427 | 3.85 | -3.4540 |
| E.2 | -3.3640 | 495 | 70 | 8.82 | 383 | -3.3736 | 460 | 185 | 9.02 | 380 | -3.2387 | 500 | 354 | 4.26 | -3.3821 |
| E.3 | -3.2889 | 485 | 87 | 8.73 | 396 | -3.2924 | 387 | 212 | 10.83 | 358 | -3.2200 | 500 | 386 | 3.81 | -3.2989 |
| E.4 | -3.4432 | 469 | 77 | 8.99 | 405 | -3.4412 | 477 | 178 | 8.45 | 347 | -3.2745 | 500 | 427 | 3.87 | -3.4540 |
| E.5 | -3.3759 | 495 | 69 | 8.83 | 397 | -3.3849 | 497 | 185 | 9.57 | 354 | -3.1959 | 500 | 403 | 3.56 | -3.3908 |

is evident and remarkable how $MSPSd\tau_k$ defeats $MSPS$ with $\tau = 0.7$ in almost all of the problems.

Fig.(7) shows the angles' distribution in radians for the problem $C.2$, which is representative of the typical behavior for all problems except E

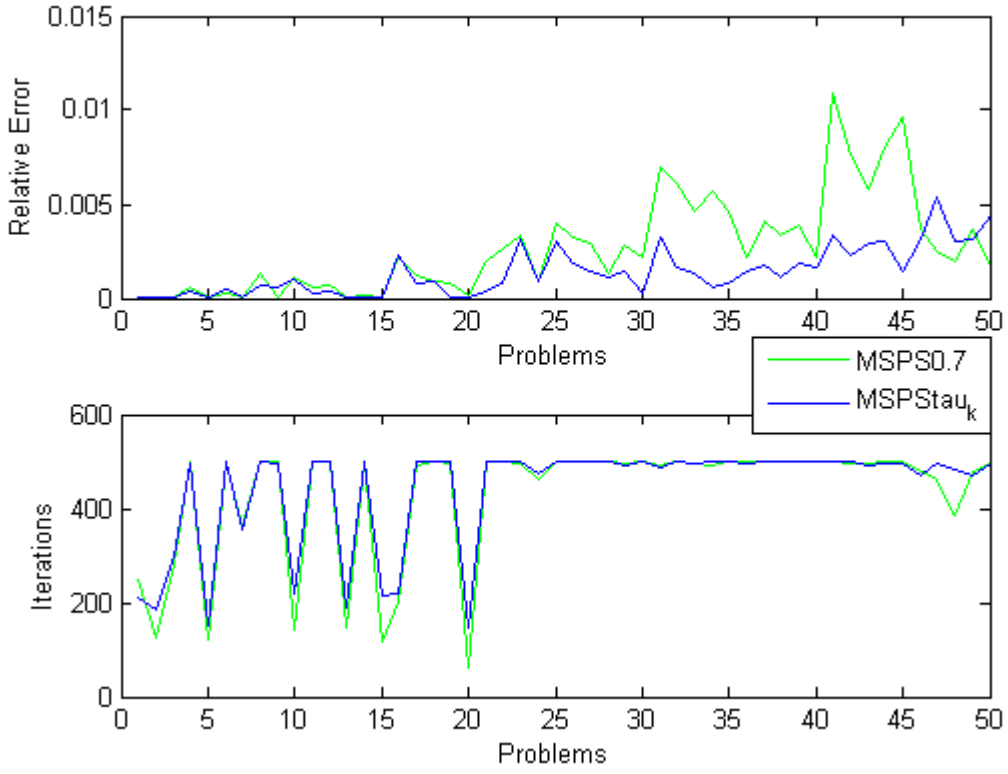


Fig. 6: Error - Iterations Comparisons

problems. We can see $\beta$ is obtuse in most of the iterations for $MSPS$ with $\tau = 0$. However, when $\tau$ is not zero $\beta$ is obtuse for a few initial iterations, then becomes acute, showing clearly the positive effect of $\tau$ in diminishing the Zigzagging of Kind I. Also, $\omega$ and $\theta$ show a steady behavior without sudden changes. Moreover, $\theta$ is typically lower than $\omega$, which suggests the direction in each iteration is kept close to its history. Notice, $MSPSd\tau_k$ and $MSPS$ with $\tau = 0.7$ show similar behavior in the angle's distribution.

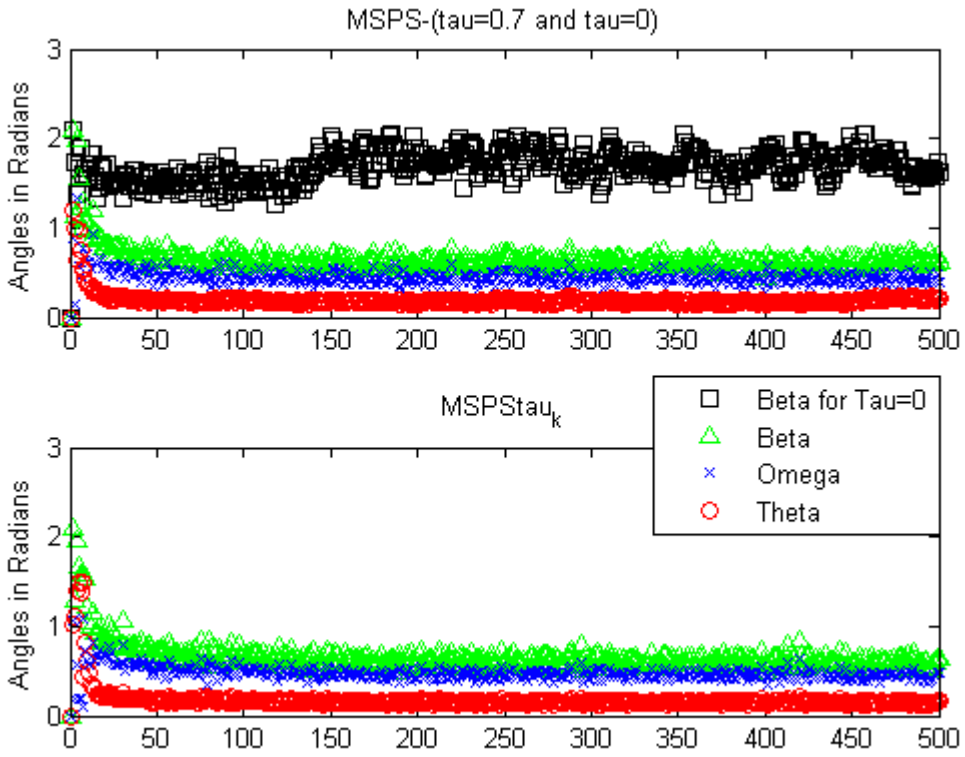


Fig. 7: C.2 - Problem

## 6 Final remarks

We carried out a comprehensive convergence analysis for the $MSPS$, and proved its convergence when the momentum parameter $\tau$ is constant and less than one.

We proposed a new heuristic for the $MSPS$ to dynamically choose the value of $\tau_k$ at each iteration to overcome Zigzagging Phenomenon of Kind I, which we called Dynamic-$\tau_k$, and presented the resulting new method $MSPSd\tau_k$. Extensive numerical experimentation on set covering problems was performed to compare $MSPS$ with no momentum term, $MSPS$ with constant momentum parameter, and $MSPSd\tau_k$. Our numerical results showed that $MSPSd\tau_k$ reaches better results and in some cases in less iterations.

Finally, proving convergence for the $MSPSd\tau_k$ is an open topic, that deserves special attention and it will be studied in future works. Likewise, new heuristics will be explored.

**Acknowledgements:** The authors would like to thank the two referees for carefully reading the manuscript and for several suggestions that improved the presentation of this work.

This work was supported by the National Science Foundation (grant DMS-1460699).

## References

[1] M. Loreto and A. Crema. Convergence analysis for the modified spectral projected subgradient method. *Optimization Letters*, 9:915–929, 2015. doi: 10.1007/s11590-014-0792-0.

[2] A. Crema, M. Loreto, and M. Raydan. Spectral projected subgradient with a momentum term for the lagrangean dual approach. *Computers and Operation Research*, 34:3174–3186, 2007. doi: 10.1016/j.cor.2005.11.024.

[3] E.G. Birgin, J.M. Martinez, and M. Raydan. Nonmonotone spectral projected gradient methods on convex set. *SIAM J. Opt.*, 10:1196–1211, 2000. doi: 10.1137/S1052623497330963.

[4] E.G. Birgin, J.M. Martinez, and M. Raydan. Algorithm 813: SPG-software for convex-constrained optimization. *ACM Transactions on Mathematical Software*, 27:340–349, 2001. doi: 10.1145/502800.502803.

[5] D. Plaut, S. Nowlan, and G.E. Hinton. Experiments on learning by back propagation. Technical Report CMU-CS-86-126, Department of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1986.

[6] B. Guta. *Subgradient Optimization Methods in Integer Programming.* VDM Verlag, Germany, 2009. ISBN 978-3639201178.

[7] L. Grippo, F. Lampariello, and S. Lucidi. A nonmonotone line search technique for Newton's method. *SIAM J. Numer. Anal.*, 23:707–716, 1986. doi: 10.1137/0723046.

[8] W. La Cruz, J.M. Martinez, and M. Raydan. Spectral residual method without gradient information for solving large-scale nonlinear systems. *Math. of Comp.*, 75:1449–1466, 2006. doi: 10.1090/S0025-5718-06-01840-0.

[9] E.G. Birging, J.M. Martinez, and M. Raydan. Spectral projected gradient methods. *Encyclopedia of Optimization, Second Ed., Editors: C. A. Floudas and P. M. Pardalos*, 19:3652–3659, 2009. ISBN 978-0-387-74760-6.

[10] E.G. Birging, J.M. Martinez, and M. Raydan. Spectral projected gradient methods: Review and perspectives. *Journal of Statistical Software*, 60:1–21, 2014. doi: 10.18637/jss.v060.i03.

[11] B.T. Polyak. A general method of solving stremum problems. *Soviet Mathematics Doklady*, 8:593–597, 1967. doi:.

[12] C.M. Bishop. *Neural Networks for Pattern Recognition*. Oxford University Press, New York, 1997. ISBN 978-0198538646.

[13] P. Tseng. An incremental gradient(-projection) method with momentum term and adaptive stepsize rule. *SIAM J. Opt.*, 8:506–531, 1998. doi: 10.1137/S1052623495294797.

[14] J.E. Beasley. Or-library: Distributing test problems by electronic mail. *Journal of Operational Research Society*, 41:1069–1072, 1990. doi: 10.1057/jors.1990.166.

[15] A. Caprara, M. Fischetti, and P. Toth. A heuristic method for the set covering problem. *Operations Research*, 47:730–743, 1999. doi: 10.1287/opre.47.5.730.

[16] M. Raydan. The Barzilai and Borwein gradient method for the large scale unconstrained minimization problem. *SIAM J. Opt.*, 7:26–33, 1997. doi: 10.1137/S1052623494266365.