Approximation algorithms for the bi-criteria weighted MAX-CUT problem

Eric Angel, Evripidis Bampis, Laurent Gourvès

Approximation Algorithms for the Bi-criteria Weighted MAX-CUT Problem

Eric Angel, Evripidis Bampis, and Laurent Gourvès

LaMI, CNRS UMR 8042, Université d'Évry Val d'Essonne, France
{angel,bampis,lgourves}@lami.univ-evry.fr

Abstract. We consider a generalization of the classical MAX-CUT problem where two objective functions are simultaneously considered. We derive some theorems on the existence and the non-existence of feasible cuts that are at the same time near optimal for both criteria. Furthermore, two approximation algorithms with performance guarantee are presented. The first one is deterministic while the second one is randomized.

1 Introduction

Given an undirected graph $G(V, E)$ with non-negative edge weights w_{ij}, the objective of the Maximum Cut problem (MAX-CUT) is to find a partition of the vertex set into two subsets S and $\overline{S}$, such that the sum of the weights of the edges having endpoints in different subsets is maximum. Formally, the weight of the cut $(S, \overline{S})$ to be maximized is given by

$$W(S, \overline{S}) = \sum_{i \in S, j \in \overline{S}} w_{ij}.$$

This well known combinatorial problem was shown to be **NP**-complete by Karp [5]. It has applications in many fields including VLSI circuit design and statistical Physics [1].

In this article, we study a *bi-criteria* version of the MAX-CUT problem. Formally, we are given an undirected graph $G(V, E)$ and two distinct weighting functions. Each feasible cut is then evaluated with respect to these two criteria.

In general no feasible solution can meet optimality simultaneously for both criteria. However, a set of solutions which *dominates*[1] all the others (the so-called *Pareto curve*) always exists. Because of the complexity of the classical (mono-criterion) MAX-CUT problem, determining this Pareto curve is computationnally problematic. Indeed, the bi-criteria MAX-CUT problem generalizes MAX-CUT. Moreover, the size of the Pareto curve, i.e. the number of non-dominated solutions, may be exponential.

[1] A solution x dominates another solution y if x is at least as good as y for all criteria and strictly better for at least one criterion.

Concerning *multi-criteria optimization* (see [2] for a recent book on the topic), three different approaches are often followed: the *budget approach*, the *Pareto curve approach* and the *simultaneous approach*. In this article we follow the third one.

By taking as a reference an *ideal solution*, namely a (not necessarily feasible) cut which simultaneously maximizes all objective functions, one tries to compute a feasible cut which approximates this ideal solution with a performance guarantee on each criterion.

In this direction, Stein and Wein [8] considered a scheduling problem with two well studied criteria, namely the *makespan* and the *average weighted completion time*. They derived existence and non-existence theorems on schedules that are simultaneously near-optimal with respect to both objective functions. A series of recent papers follow this approach [7, 9–12].

In this article, we follow the same approach for the bi-criteria MAX-CUT problem. The paper is organized as follows: A formal presentation of the problem is given in Section 2. Sections 3 and 4 are respectively devoted to a deterministic and a randomized bi-criteria approximation algorithm with performance guarantee. Finally, some outlooks and concluding remarks are given in Section 5.

2 Formalization and notation

We are given an undirected graph $G(V, E)$ where each edge $e \in E$ has a non-negative weight w_e and a non-negative length l_e. A solution $(S, \overline{S})$ is feasible if it constitutes a partition of V. An edge e belongs to a cut $(S, \overline{S})$, denoted by $e \in (S, \overline{S})$, if e links a vertex in S and a vertex in $\overline{S}$. The following objective functions, namely the total weight and the total length, are considered:

$$W(S, \overline{S}) = \sum_{e \in (S, \overline{S})} w_e \text{ and } L(S, \overline{S}) = \sum_{e \in (S, \overline{S})} l_e.$$

Let $(O, \overline{O})$ (resp. $(P, \overline{P})$) be a feasible cut which maximizes the total weight (resp. length). Let $(I, \overline{I})$ be an *ideal* (not necessarily feasible) cut such that:

$$W(I, \overline{I}) = W(O, \overline{O}) = OPTW \text{ and } L(I, \overline{I}) = L(P, \overline{P}) = OPTL.$$

The bi-criteria weighted MAX-CUT problem is then to find a feasible cut $(A, \overline{A})$ such that:

$$W(A, \overline{A}) \geq \alpha \, OPTW \text{ and } L(A, \overline{A}) \geq \beta \, OPTL$$

where $0 < \alpha \leq 1$ and $0 < \beta \leq 1$. An (α, β)-approximation algorithm outputs a solution which is simultaneously α-approximate on the first criterion (the total weight) and β-approximate on the second criterion (the total length).

3 A deterministic approximation algorithm

Given a deterministic α-approximation algorithm **Al** for the mono-criterion weighted MAX-CUT problem, one can build an $(\alpha/2, \alpha/2)$-approximation algorithm for the bi-criteria weighted MAX-CUT problem. The algorithm called **Bi-Approx** follows:

```
Bi-Approx
Input:   $G$ and Al
Step 1:  Find $(S_1, \overline{S_1})$ with Al s.t. $W(S_1, \overline{S_1}) \geq \alpha.OPTW$
Step 2:  Find $(S_2, \overline{S_2})$ with Al s.t. $L(S_2, \overline{S_2}) \geq \alpha.OPTL$
Step 3:  Build $(S_3, \overline{S_3})$ s.t. $S_3 = (S_1 \cap S_2) \cup (\overline{S_1} \cap \overline{S_2})$
Step 4:  If $L(S_1, \overline{S_1}) \geq 0.5\, L(S_2, \overline{S_2})$
         Then Return $(S_1, \overline{S_1})$
         Else If $W(S_2, \overline{S_2}) \geq 0.5\, W(S_1, \overline{S_1})$
             Then Return $(S_2, \overline{S_2})$
             Else Return $(S_3, \overline{S_3})$
```

Theorem 1. **Bi-Approx** *is a deterministic* $(\alpha/2, \alpha/2)$*-approximation algorithm for the bi-criteria weighted* MAX-CUT *problem if* **Al** *is a deterministic* α*-approximation algorithm for the mono-criterion weighted* MAX-CUT *problem.*

Proof. Clearly, if **Bi-Approx** returns $(S_1, \overline{S_1})$ or $(S_2, \overline{S_2})$ then the solution returned is either $(\alpha, \alpha/2)$ or $(\alpha/2, \alpha)$-approximate, and hence $(\alpha/2, \alpha/2)$-approximate. In the following, we suppose that $(S_3, \overline{S_3})$ is returned by **Bi-Approx** and we prove that it is an $(\alpha/2, \alpha/2)$-approximate cut.

We partition V into four subsets X, Y, Z and T such that $(S_1, \overline{S_1}) = (X \cup Y, Z \cup T)$ and $(S_2, \overline{S_2}) = (X \cup Z, Y \cup T)$. Vertices of each subset are shrinked into *super-nodes* denoted by v_X, v_Y, v_Z and v_T. More precisely, all nodes $v \in X$ fall into v_X, all nodes $v \in Y$ fall into v_Y etc. Edges between two super-nodes are also shrinked into one *super-edge* such that:

$$w_{v_A\, v_B} = \sum_{v \in A, v' \in B} w_{v\, v'} \text{ and } l_{v_A\, v_B} = \sum_{v \in A, v' \in B} l_{v\, v'}$$

where $A \in \{X, Y, Z, T\}$, $B \in \{X, Y, Z, T\}$ and $A \neq B$. Finally, we get a new graph K_4 as depicted in Figure 2.

Now observe that if $l_{v_X\, v_T} + l_{v_Y\, v_Z} \geq l_{v_X\, v_Y} + l_{v_Z\, v_T}$ is true then we get a contradiction since instead of $(S_3, \overline{S_3})$, $(S_1, \overline{S_1})$ would have been returned:

$$\begin{aligned} l_{v_X\, v_T} + l_{v_Y\, v_Z} &\geq l_{v_X\, v_Y} + l_{v_Z\, v_T} \\ l_{v_X\, v_T} + l_{v_Y\, v_Z} &\geq (l_{v_X\, v_Y} + l_{v_Z\, v_T} + l_{v_X\, v_T} + l_{v_Y\, v_Z})/2 \\ L(S_1, \overline{S_1}) &\geq L(S_2, \overline{S_2})/2 \end{aligned}$$

Symmetrically, if $w_{v_X\, v_T)} + w_{v_Y\, v_Z} \geq w_{v_X\, v_Z} + w_{v_Y\, v_T}$ is true then we get a contradiction since instead of $(S_3, \overline{S_3})$, $(S_2, \overline{S_2})$ would have been returned:

$$\begin{aligned} w_{v_X\, v_T} + w_{v_Y\, v_Z} &\geq w_{v_X\, v_Z} + w_{v_Y\, v_T} \\ w_{v_X\, v_T} + w_{v_Y\, v_Z} &\geq (w_{v_X\, v_Z} + w_{v_Y\, v_T} + w_{v_X\, v_T} + w_{v_Y\, v_Z})/2 \\ W(S_2, \overline{S_2}) &\geq W(S_1, \overline{S_1})/2 \end{aligned}$$

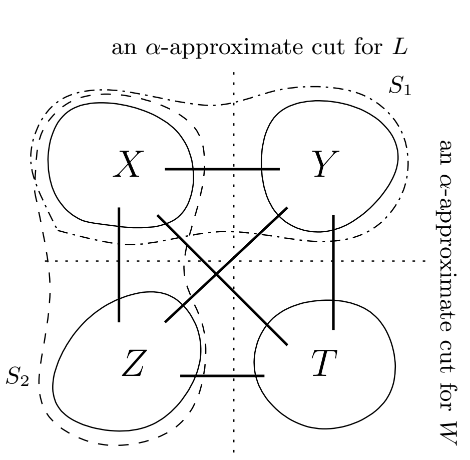
an α-approximate cut for L
S_1
X
Y
Z
T
S_2
an α-approximate cut for W

Fig. 1. Vertices of G are partitioned into four subsets X, Y, Z and T. This partition depends on $(S_1, \overline{S_1})$ and $(S_2, \overline{S_2})$.

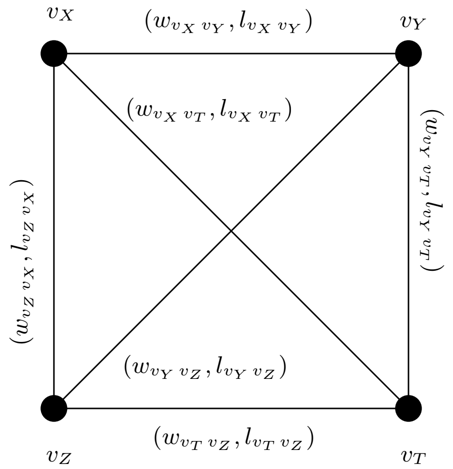
v_X
$(w_{v_X\, v_Y}, l_{v_X\, v_Y})$
v_Y
$(w_{v_X\, v_T}, l_{v_X\, v_T})$
$(w_{v_Z\, v_X}, l_{v_Z\, v_X})$
$(w_{v_Y\, v_T}, l_{v_Y\, v_T})$
$(w_{v_Y\, v_Z}, l_{v_Y\, v_Z})$
$(w_{v_T\, v_Z}, l_{v_T\, v_Z})$
v_Z
v_T

Fig. 2. Vertices and edges of G are shrinked to get a complete graph with four nodes.

Thus we have:

$$l_{v_X\, v_T} + l_{v_Y\, v_Z} < l_{v_X\, v_Y} + l_{v_Z\, v_T} \text{ and} \quad (1)$$

$$w_{v_X\, v_T} + w_{v_Y\, v_Z} < w_{v_X\, v_Z} + w_{v_Y\, v_T}. \quad (2)$$

From inequality (1) we get:

$$\begin{aligned}
(l_{v_X\, v_Y} + l_{v_Z\, v_T})/2 &> (l_{v_X\, v_T} + l_{v_Y\, v_Z})/2 \\
l_{v_X\, v_Z} + l_{v_Y\, v_T} + (l_{v_X\, v_Y} + l_{v_Z\, v_T})/2 &> (l_{v_X\, v_T} + l_{v_Y\, v_Z})/2 \\
l_{v_X\, v_Z} + l_{v_Y\, v_T} + l_{v_X\, v_Y} + l_{v_Z\, v_T} &> (l_{v_X\, v_T} + l_{v_Y\, v_Z} + \\
&\quad + l_{v_X\, v_Y} + l_{v_Z\, v_T})/2 \\
L(S_3, \overline{S_3}) &> 0.5 L(S_2, \overline{S_2}) \\
L(S_3, \overline{S_3}) &\geq \frac{\alpha}{2} OPTL
\end{aligned}$$

From inequality (2) we get:

$$\begin{aligned}
(w_{v_X\, v_Z} + w_{v_Y\, v_T})/2 &> (w_{v_X\, v_T} + w_{v_Y\, v_Z})/2 \\
w_{v_X\, v_Y} + w_{v_Z\, v_T} + (w_{v_X\, v_Z} + w_{v_Y\, v_T})/2 &> (w_{v_X\, v_T} + w_{v_Y\, v_Z})/2 \\
w_{v_X\, v_Y} + w_{v_Z\, v_T} + w_{v_X\, v_Z} + w_{v_Y\, v_T} &> (w_{v_X\, v_T} + w_{v_Y\, v_Z} + \\
&\quad + w_{v_X\, v_Z} + w_{v_Y\, v_T})/2 \\
W(S_3, \overline{S_3}) &> 0.5 W(S_1, \overline{S_1}) \\
W(S_3, \overline{S_3}) &> \frac{\alpha}{2} OPTW
\end{aligned}$$

□

The analysis of **Bi-Approx** is tight. To see it, consider the instance given in Figure 3 where K is a large integer. The ideal point has a total weight and a total length equal to 1 while $(S_1, \overline{S_1})$ achieves the values $(\alpha, \alpha\frac{K-1}{2K})$ and $(S_2, \overline{S_2})$ achieves the values $(\alpha\frac{K-1}{2K}, \alpha)$. The algorithm returns a solution $(S_3, \overline{S_3})$ such that $S_3 = \{v_1, v_3, v_5\}$ and its total weight and total length are both equal to $\alpha\frac{K+1}{2K}$. When K tends to infinity, the solution returned tends to be $(\alpha/2, \alpha/2)$-approximate.

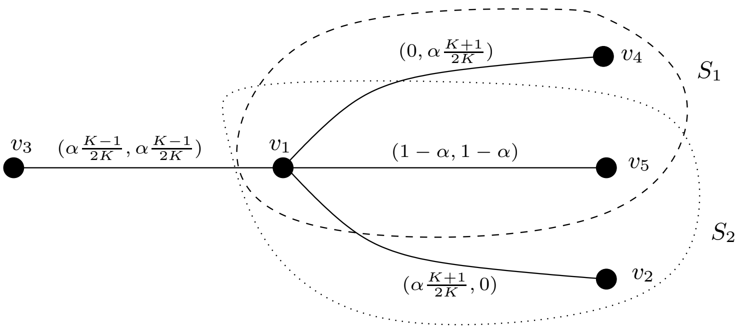
$(0, \alpha\frac{K+1}{2K})$
v_4
S_1
v_3
$(\alpha\frac{K-1}{2K}, \alpha\frac{K-1}{2K})$
v_1
$(1-\alpha, 1-\alpha)$
v_5
S_2
v_2
$(\alpha\frac{K+1}{2K}, 0)$

Fig. 3. Instance for which **Bi-Approx** returns an $(\alpha/2, \alpha/2)$-approximate solution.

Corollary 1. *There exists a deterministic* $(0.43928, 0.43928)$*-approximate algorithm for the bi-criteria weighted* MAX-CUT *problem.*

Proof. Replace **Al** in **Bi-Approx** by the derandomized algorithm of Goemans and Williamson [3] which is a 0.87856-approximate algorithm and the result follows. □

Interestingly, an existence theorem can be derived from the algorithm **Bi-Approx**.

Theorem 2. *For all instances of the bi-criteria weighted* MAX-CUT *problem, there always exists a feasible solution which approximates the ideal point within a ratio* $1/2$ *on the two criteria.*

Proof. Suppose that **Al** in **Bi-Approx** is an optimal (1-approximate) algorithm for the mono-criterion weighted MAX-CUT problem and the result follows. □

The question whether the above theorem can be improved arises but the following theorem brings a negative answer.

Theorem 3. *No* (α, β)*-approximation algorithm with* $\alpha > 1/2$ *or* $\beta > 1/2$ *is likely to exist for the bi-criteria* MAX-CUT *problem.*

Proof. Consider the complete graph K_3 whose edges e, e' and e'' are such that $w_e = l_{e'} = 0$ and $l_e = w_{e'} = w_{e''} = l_{e''} = 1$. The ideal solution $(I, \overline{I})$ has a total weight and a total length both equal to 2 while no feasible cut has a total weight and a total length simultaneously strictly superior to 1. □

4 A randomized approximation algorithm

As usual, we consider that a randomized algorithm for a mono-criterion maximization problem is an α-expected approximate algorithm if the expected value (denoted by $E[X]$) of the solution returned is at least α times the value (denoted by OPT) of an optimal solution: $E[X] \geq \alpha OPT$.

When randomization is considered, the bi-criteria weighted MAX-CUT problem is then to find a feasible cut $(A, \overline{A})$ such that $E[W(A, \overline{A})] \geq \alpha OPTW$ and $E[L(A, \overline{A})] \geq \beta OPTL$ where $0 < \alpha \leq 1$ and $0 < \beta \leq 1$.

There is no hope to get an (α, β)-expected approximate algorithm for the bi-criteria weighted MAX-CUT problem with $\alpha = \beta$ and $\alpha > 2/3$. To see it, consider the example given in Figure 4 where the ideal cut $(I, \overline{I})$ achieves the values $(1, 1)$. Four cuts $(S_1, \overline{S_1})$, $(S_2, \overline{S_2})$, $(S_3, \overline{S_3})$ and $(S_4, \overline{S_4})$ are feasible with values respectively $(0, 0)$, $(2/3, 2/3)$, $(1/3, 1)$, and $(1, 1/3)$. Let **Ran Al** be a randomized algorithm which outputs $(S_i, \overline{S_i})$ with a probability p_i. Obviously, one has $p_1 + p_2 + p_3 + p_4 = 1$. The expected value of the cut $(S, \overline{S})$ output by **Ran Al** is:

$$E[W(S, \overline{S})] = \frac{2p_2}{3} + \frac{p_3}{3} + p_4 \text{ and } E[L(S, \overline{S})] = \frac{2p_2}{3} + p_3 + \frac{p_4}{3}.$$

The problem is then to find p_1, p_2, p_3 and p_4 such that $E[W(S, \overline{S})] \geq \alpha$, $E[L(S, \overline{S})] \geq \alpha$ and α is maximized. When $p_1 = p_3 = p_4 = 0$ and $p_2 = 1$, α reaches $2/3$ which is the best possible value. As a consequence, no randomized algorithm can be (α, α)-expected approximate with $\alpha > 2/3$.

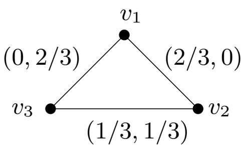

Fig. 4. The ideal cut $(I, \overline{I})$ has a total weight and a total length both equal to 1.

This statement has a consequence in the approximability of the weighted bi-criteria MAX-CUT problem. Indeed, there is no hope to design a deterministic (α, β)-approximate algorithm such that $\alpha + \beta > 4/3$. To see it, suppose that we have such an algorithm. One can build two solutions $(S_1, \overline{S_1})$ and $(S_2, \overline{S_2})$ such that $W(S_1, \overline{S_1}) \geq \alpha OPTW$, $L(S_1, \overline{S_1}) \geq \beta OPTL$, $W(S_2, \overline{S_2}) \geq \beta OPTW$ and $L(S_2, \overline{S_2}) \geq \alpha OPTL$. Now consider the randomized algorithm which consists in returning $(S_1, \overline{S_1})$ with a probability $1/2$ and $(S_2, \overline{S_2})$ with a probability $1/2$. We would get an $(\frac{\alpha+\beta}{2}, \frac{\alpha+\beta}{2})$-expected approximate solution $(S, \overline{S})$ and $\frac{\alpha+\beta}{2} > 2/3$.

The algorithm (called **Ransam** in [4]) which consists in building a cut $(S, \overline{S})$ by puting equiprobably a vertex $v \in V$ to either S or $\overline{S}$ is $1/2$-expected approximate for the mono-criterion weighted MAX-CUT problem. One can remark that it achieves the same performance guarantee for a multi-criteria weighted MAX-CUT problem. However, a better randomized algorithm can be built for the bi-criteria MAX-CUT problem. We propose an algorithm called **Ran Bi-Approx** which uses a mono-criterion α-approximation algorithm (called **Al** in the following).

```
Ran Bi-Approx
Input: $G$ and Al
Step 1: Find $(S_1, \overline{S_1})$ with Al s.t. $W(S_1, \overline{S_1}) \geq \alpha\, OPTW$
Step 2: Find $(S_2, \overline{S_2})$ with Al s.t. $L(S_2, \overline{S_2}) \geq \alpha\, OPTL$
Step 3: Build $(S_3, \overline{S_3})$ s.t. $S_3 = (S_1 \cap S_2) \cup (\overline{S_1} \cap \overline{S_2})$
Step 4: Let $\gamma = (3 - \sqrt{5})/2$
Step 5: If $W(S_2, \overline{S_2}) \geq \gamma W(S_1, \overline{S_1})$
            Then If $L(S_1, \overline{S_1}) \geq \gamma L(S_2, \overline{S_2})$
                Then Return $(S_1, \overline{S_1})$ with a probability 0.5
                    and $(S_2, \overline{S_2})$ with a probability 0.5
                Else Return $(S_1, \overline{S_1})$ with a probability $\gamma$
                    and $(S_2, \overline{S_2})$ with a probability $1 - \gamma$
            Else If $L(S_1, \overline{S_1}) \geq \gamma L(S_2, \overline{S_2})$
                Then Return $(S_1, \overline{S_1})$ with a probability $1 - \gamma$
                    and $(S_2, \overline{S_2})$ with a probability $\gamma$
                Else Return $(S_3, \overline{S_3})$
```

Theorem 4. Ran Bi-Approx *is a randomized* $(\frac{\sqrt{5}-1}{2}\alpha, \frac{\sqrt{5}-1}{2}\alpha)$*-expected approximation algorithm for the bi-criteria weighted* MAX-CUT *problem if* **Al** *is an* α*-approximation algorithm.*

Proof. The algorithm considers four cases. For the first case, we suppose that:

$$W(S_2, \overline{S_2}) \geq \gamma W(S_1, \overline{S_1}) \text{ and } L(S_1, \overline{S_1}) \geq \gamma L(S_2, \overline{S_2}).$$

So, we have:

$$W(S_2, \overline{S_2}) \geq \gamma\alpha OPTW \text{ and } L(S_1, \overline{S_1}) \geq \gamma\alpha OPTL.$$

Since the solution returned in this case is $(S_1, \overline{S_1})$ with a probability 0.5 and $(S_2, \overline{S_2})$ with a probability 0.5, the expected value on each criterion of the solution returned is at least $\frac{\alpha(1+\gamma)}{2}$ times the optimum.

For the second case, we suppose that:

$$W(S_2, \overline{S_2}) \geq \gamma W(S_1, \overline{S_1}) \text{ and } L(S_1, \overline{S_1}) \geq 0.$$

So, we have $W(S_2, \overline{S_2}) \geq \gamma\alpha OPTW$. Since the solution returned in this case is $(S_1, \overline{S_1})$ with a probability $\gamma = \frac{1-\gamma}{2-\gamma}$ and $(S_2, \overline{S_2})$ with a probability $1-\gamma = \frac{1}{2-\gamma}$, the expected value on each criterion of the solution returned is at least $\frac{\alpha}{2-\gamma}$ times the optimum.

The third case is symmetric to the second case, the expected value on each criterion of the solution returned is at least $\frac{\alpha}{2-\gamma}$ times the optimum.

For the fourth case, we suppose that:

$$W(S_2,\overline{S_2}) < \gamma W(S_1,\overline{S_1}) \text{ and } L(S_1,\overline{S_1}) < \gamma L(S_2,\overline{S_2}).$$

As it was done before, we consider that the set of vertices is partitioned into four subsets (see Figure 1) and the proof is done on a simple K_4 graph (see Figure 2). So, we have:

$$\begin{aligned} w_{v_X\,v_Y} + w_{v_Z\,v_T} + w_{v_X\,v_T} + w_{v_Y\,v_Z} &< \gamma\big(w_{v_X\,v_Z} + w_{v_Y\,v_T} + \\ &\quad + w_{v_X\,v_T} + w_{v_Y\,v_Z}\big) \qquad (3) \\ l_{v_X\,v_Z} + l_{v_Y\,v_T} + l_{v_X\,v_T} + l_{v_Y\,v_Z} &< \gamma\big(l_{v_X\,v_Y} + l_{v_Z\,v_T} + \\ &\quad + l_{v_X\,v_T} + l_{v_Y\,v_Z}\big). \qquad (4) \end{aligned}$$

From inequality (3), we get:

$$\begin{aligned} w_{v_X\,v_T} + w_{v_Y\,v_Z} &< \gamma\big(w_{v_X\,v_Z} + w_{v_Y\,v_T} + \\ &\quad + w_{v_X\,v_T} + w_{v_Y\,v_Z}\big) \\ (1-\gamma)\big(w_{v_X\,v_T} + w_{v_Y\,v_Z}\big) &< \gamma\big(w_{v_X\,v_Z} + w_{v_Y\,v_T}\big) \\ \frac{(1-\gamma)}{\gamma}\big(w_{v_X\,v_T} + w_{v_Y\,v_Z}\big) &< w_{v_X\,v_Z} + w_{v_Y\,v_T} \\ \frac{(1-\gamma)}{\gamma}\big(w_{v_X\,v_T} + w_{v_Y\,v_Z} + w_{v_X\,v_Z} + w_{v_Y\,v_T}\big) &< \frac{1}{\gamma}\big(w_{v_X\,v_Z} + w_{v_Y\,v_T}\big) \\ (1-\gamma)\big(w_{v_X\,v_T} + w_{v_Y\,v_Z} + w_{v_X\,v_Z} + w_{v_Y\,v_T}\big) &< w_{v_X\,v_Z} + w_{v_Y\,v_T} + \\ &\quad + w_{v_X\,v_Y} + w_{v_Z\,v_T} \\ (1-\gamma)W(S_1,\overline{S_1}) &< W(S_3,\overline{S_3}) \end{aligned}$$

Symmetrically, from inequality (4) we get:

$$(1-\gamma)L(S_2,\overline{S_2}) < L(S_3,\overline{S_3})$$

In this case, $(S_3,\overline{S_3})$ is returned and its value on each criterion is at least $(1-\gamma)\alpha$ times the optimum.

Let $f(\gamma) = \min\{1-\gamma, \frac{1}{2-\gamma}, \frac{1+\gamma}{2}\}$ for $0 \leq \gamma \leq 1$. This function finds its maximum when $\gamma = \frac{3-\sqrt{5}}{2}$. As a consequence, the solution returned by **Ran Bi-Approx** has an expected value on each criterion which is at least $\frac{\sqrt{5}-1}{2}\alpha$ times the optimum. □

Corollary 2. *There exists a randomized* $(0.54297, 0.54297)$*-expected approximate algorithm for the bi-criteria weighted* MAX-CUT *problem.*

Proof. Replace **Al** by the algorithm of Goemans and Williamson [3] in **Ran Bi-Approx** and the result follows. □

5 Concluding remarks

Since we considered a bi-criteria MAX-CUT problem and provided approximation algorithms, the question whether it is possible to get similar results with more than two criteria arises. Unfortunately, the example given in Figure 5 shows that it is not possible to build a deterministic algorithm which approximates the ideal point with a performance guarantee when tree criteria are considered. As a consequence, there is no hope to find an approximation algorithm with performance guarantee for the k-criteria weighted MAX-CUT problem where $k > 2$. However, **Ransam** remains a 1/2-expected approximation algorithm for any k-criteria weighted MAX-CUT problem.

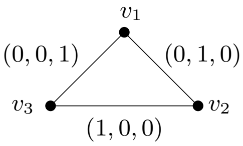

Fig. 5. The ideal cut $(I,\overline{I})$ achieves the values $(1,1,1)$ while any feasible cut achieves 0 on at least one coordinate. Thus, no approximation factor can be guaranteed.

Remark that approximation results for the k-criteria weighted MAX-CUT problem can be found if another approach is considered. Indeed, if we restrict ourselves to feasible solutions then rarely a solution will dominate all the others (i.e. will be better than the others on each criterion) but a set of solutions which dominates all the others always exists. This set of solutions is called the *Pareto curve* and Papadimitriou and Yannakakis [6] proved that an approximation with performance guarantee of this curve (an *ε-approximate Pareto curve*) always exists.

Acknowledgement: We thank Martin Skutella for giving us the example of Figure 3.

References

1. F. Barahona, M. Grötschel, M. Jünger and G. Reinelt. An Application of Combinatorial Optimization to Statistical Physics and Circuit Layout Design. *Operations Research*, 36:493–513, 1998.
2. M. Ehrgott. Multicriteria optimization, Lecture Notes in Economics and Mathematical Systems N 491. Springer, 2000.
3. M.X. Goemans and D.P. Williamson. Improved Approximation Algorithms for Maximum Cut and Satisfiability Problems Using Semidefinite Programming, *Journal of the ACM*, 42(6), 1115–1145, 1995.
4. J. Hromkovič. Algorithmics for Hard Problems: Introduction to Combinatorial Optimization, Randomization, Approximation, and Heuristics. Springer, 2001.

5. R.M. Karp. Reducability Among Combinatorial Problems. In R. Miller and J. Thatcher (editors), *Complexity of Computer Computations*, pp 85–103, Plenum Press, NY, 1972.
6. C.H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources, In *Proceedings of th 41th Annual IEEE Symposium on Foundations of Computer Science*, 86–92, 2000.
7. A. Rasala, C. Stein, E. Torng and P. Uthaisombut. Existence theorems, lower bounds and algorithms for scheduling to meet two objectives, In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, 723–731, 2002.
8. C. Stein and J. Wein. On the existence of schedules that are near-optimal for both makespan and total weighted completion time, *Operations Research Letters*, 21, 115–122, 1997.
9. J. Aslam and A. Rasala and C. Stein and N. Young, Improved bicriteria existence theorems for scheduling, *Proceedings 10th Annual ACM-SIAM Symposium on Discrete Algorithms*, 846–847, 1999.
10. E. Angel and E. Bampis and A.V. Fishkin, A note on scheduling to meet two min-sum objectives, *Proceeding 9th International Workshop on Project Management and Scheduling (PMS'2004)*, 143–146, 2004.
11. F. Baille and E. Bampis and C. Laforest, A Note on Bicriteria Schedules with Optimal Approximations Ratios, *Parallel Processing Letters*, 315–323, 14, 2004.
12. E. Bampis and A. Kononov", Bicriteria approximation algorithms for scheduling problems with communications delays, *Annals of Operations Research*, 2005, à paraître.