PUBLICATIONS DE L'INSTITUT MATHÉMATIQUE
Nouvelle série, tome 99(113) (2016), 51–58 DOI: 10.2298/PIM1613051I

IMPROVED MIXED INTEGER LINEAR PROGRAMING FORMULATIONS FOR ROMAN DOMINATION PROBLEM

Marija Ivanović

Abstract. The Roman domination problem is considered. An improvement of two existing Integer Linear Programing (ILP) formulations is proposed and a comparison between the old and new ones is given. Correctness proofs show that improved linear programing formulations are equivalent to the existing ones regardless of the variables relaxation and usage of lesser number of constraints.

1. Introduction

Domination on a graph has been extensively studied in the literature. Many variations and generalizations of this problem arose. One of them, with historical significants, was called the Roman domination problem. The Roman domination problem dates from the 4th century, when the Emperor of Rome, Constantine the Great, in order to defend an entire Empire, decreed that two types of legions should be placed in Roman provinces. The first type of legion were highly trained mobile warriors and therefore, in order to defend a province against any attack, they could move fast from one province to another. The second type of legion behaved as a local militia and were permanently stationed in a given province. The Emperor decreed that no mobile legion could ever leave a province in order to defend another one, if such an act would leave an originating province undefended.

The Roman Empire can be illustrated by an undirected graph where each province is represented by a different vertex. Two vertices will be set as adjacent if a connection between corresponding provinces exists. For connected provinces it will be said that they are neighbors. Any legion could move only over connected provinces, i.e., legion could move only along the edge of two adjacent vertices. Further, a province will be considered as defended if there is at least one legion

---

2010 *Mathematics Subject Classification:* 65K05, 90C11, 90C05, 94C15, 68R10.

Key words and phrases: Roman domination in graphs, combinatorial optimization, mixed integer linear programming.

Partially supported by the Serbian Ministry of Education, Science and Technological Developments, grant TR36006.

Communicated by Slobodan K. Simić.

stationed within it. A province without a stationed legion will be considered to be defended if a vertex which represents it is directly connected to a vertex which represents a province with two stationed legions: if there are two legions in a neighbor's province, one legion will be considered as mobile and therefore, a certain province will be considered as defended because a mobile legion could arrive fast in order to defend it. Otherwise a province is left to be undefended. For a detailed illustration see [**8**, p. 586].

The proposed problem is illustrated on a small example where the Emperor of Rome, Constantine the Great, had at his disposal only four legions to be placed and eight provinces to be defended, see Figure 1.

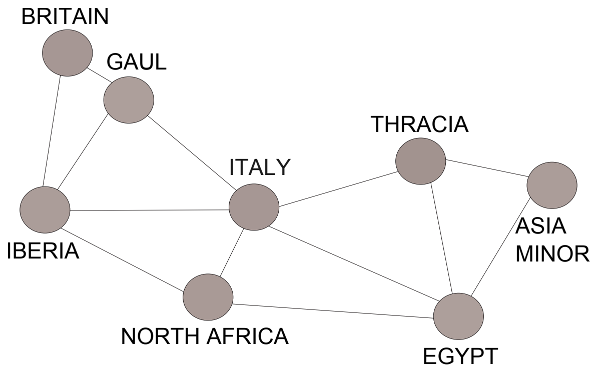

FIGURE 1. Representation of the Roman Empire illustrated on a graph.

Assigning two legions to Italy and another two to Thracia, one province was left to be undefended. Given that, in order to defend Britain, one mobile legion should move from Italy to Gaul waiting for another mobile legion to come from Thracia and then proceed to Britain. It is obvious that such strategy is not optimal, i.e. by assigning two legions to Iberia and another two to Egypt, the entire Empire will be defended. The same result could be reached also by assigning two legions to Britain or to Gaul and another two to Egypt. Note that the minimal number of legions necessary to defend the given Empire of Rome is four.

The Roman domination problem (RDP), introduced by Ian Stewart in [**10**], can be described as a problem of finding the minimal number of legions such that the entire Empire of Rome is safe.

There are several papers on this problem. The first additional developments of the RDP were proposed by ReVelle and Rosing in [**8**], while some of the most recent theoretical developments were given in [**1**, **3**, **5**–**7**].

Some special classes of graphs, such as interval graphs, intersection graphs, co-graphs and distance-hereditary graphs can be solved in linear time [**5**] but, in a general case, the Roman domination problem is NP-hard, [**4**, **5**, **9**].

This paper is organized as follows: the definition of the Roman domination problem is given in Section 2 and in Section 3, as proposed in the literature, two existing ILP formulations for solving the Roman domination problem will be reviewed. Subsequently, new alternative formulations together with the proofs of their correctness will be presented in Section 4. Conclusion, outlook on the future work and literature are given in the two final sections.

2. Problem definition

Let $G = (V, E)$ represents a finite and undirected graph with a vertex set V such that each vertex $v \in V$ represents a province and each edge, $e \in E$, represents an existing road between two adjacent provinces. There will be no loops nor multiple edges between two adjacent vertices. Let us define a neighborhood set N_v ($N_v \subset V$), of a vertex $v \in V$, such that each vertex $w \in N_v$ is adjacent to a vertex v. Finally, let us define a function $f : V \to \{0, 1, 2\}$ such that $f(v)$ is equal to a number of legions assigned to a province represented by a vertex v. Function f has to satisfy the condition that for every vertex $v \in V$ such that $f(v) = 0$ there exists a vertex $w \in N_v$ such that $f(w) = 2$. In other words, if there is an undefended province v, then there exists at least one province w, $w \in N_v$ with two stationed legions.

A small illustration of the Roman domination problem follows.

Example 2.1. Let as assume that there are 25 provinces to be defended and that each province can be represented by a particular vertex of a grid graph $G_{5,5}$. An adjacency matrix of a given graph is defined in such a way that it reflects a connection between the provinces illustrated on a Figure 2 (left).

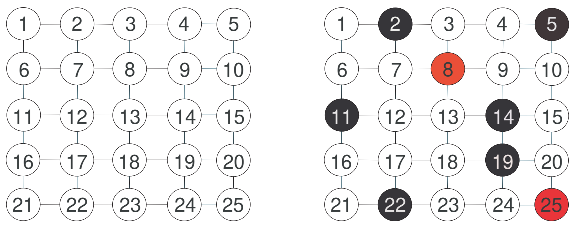

Figure 2. Illustration of Example 1.

The solution to the given problem is illustrated by coloring vertices in three colors, see Figure 2 (right) and is obtained by mathematical formulations described in Section 3. The vertices colored in black represent provinces with two assigned legions, colored in red represent provinces with one legion assigned and colored in white otherwise. Note that the entire area of 25 provinces could be defended by 14 legions, i.e. $f(v_2^*) = 2$ for all $v_2^* \in \{2, 5, 11, 14, 19, 22\}$, $f(v_1^*) = 1$ for all $v_1^* \in \{8, 25\}$ and $f(v_0^*) = 0$ for all $v_0^* \in \{1, 3, 4, 6, 7, 9, 10, 12, 13, 15, 16, 17, 18, 20, 21, 23, 24\}$.

3. Existing integer linear programing formulations

sec1 There are two ILP formulations known from the literature. The first formulation was introduced by ReVelle and Rosing in [**8**] and will be referred to as

$\mathcal{RR}$, while the second formulation was introduced by Burger at el. in [**2**] and will be referred to as $\mathcal{BVV}$.

3.1. $\mathcal{RR}$ formulation. For a function f, $f : V \to \{0, 1, 2\}$, and $i \in V$, let us define the variables

$$x_i = \begin{cases} 1, & f(i) \geqslant 1 \\ 0, & \text{otherwise} \end{cases} \qquad y_i = \begin{cases} 1, & f(i) = 2 \\ 0, & \text{otherwise.} \end{cases}$$

The $\mathcal{RR}$ formulation of the RDP can be described by

$$\min \sum_{i \in V} x_i + \sum_{i \in V} y_i \tag{3.1}$$

$$x_i + \sum_{j \in N_i} y_j \geqslant 1, \quad i \in V \tag{3.2}$$

$$y_i \leqslant x_i, \quad i \in V \tag{3.3}$$

$$x_i, y_i \in \{0, 1\}, \quad i \in V \tag{3.4}$$

The objective function value, given by (3.1), represents the number of legions used in defense. Constraints (3.2) ensure that each province is safe or there is at least one province in its neighborhood with two legions within it. By the constraints (3.3) it is ensured that provinces with two legions are safe. Decision variables x_i and y_i are preserved to be binary by the constraints (3.4).

The $\mathcal{RR}$ formulation consists of $2|V|$ binary variables and $2|V|$ constraints.

3.2. $\mathcal{BVV}$ formulation. For a function f, $f : V \to \{0, 1, 2\}$, let

$$x_i = \begin{cases} 1, & f(i) = 1 \\ 0, & \text{otherwise} \end{cases} \qquad y_i = \begin{cases} 1, & f(i) = 2 \\ 0, & \text{otherwise.} \end{cases}$$

The $\mathcal{BVV}$ formulation of the RDP can now be described by

$$\min \sum_{i \in V} x_i + 2 \sum_{i \in V} y_i \tag{3.5}$$

$$x_i + y_i + \sum_{j \in N_i} y_j \geqslant 1, \qquad i \in V \tag{3.6}$$

$$x_i + y_i \leqslant 1, \qquad i \in V \tag{3.7}$$

$$x_i, y_i \in \{0, 1\}, \quad i \in V \tag{3.8}$$

The objective function value is given by (3.5). By conditions (3.6) it is obtained that an undefended province has to be in the neighborhood of at least one province with two assigned legions. By conditions (3.7) it is given that if a province is defended by two legions then there is no need to say that it is defended by one legion as well and vice versa. Again, the decision variables x_i and y_i are preserved to be binary by the constraints (3.8).

The $\mathcal{BVV}$ formulation also consists of $2|V|$ binary variables and $2|V|$ constraints.

4. Alternative linear programing formulations

4.1. New improved $\mathcal{RR}$ formulation. Let us define

$$x_i \in [0, +\infty), y_i \in \{0, 1\}, \qquad i \in V. \tag{4.1}$$

Considering the $\mathcal{RR}$ formulation it can be noted that the binary variables x_i can be relaxed to be real. Let us mark the $\mathcal{RR}$ formulation with the given relaxation (4.1) as $\mathcal{RR}_{Imp}$. Given that, the existing $\mathcal{RR}$ formulation, which is ILP formulation, can be relaxed to the MILP $\mathcal{RR}_{Imp}$ formulation.

Theorem 4.1. *Optimal objective function value of the $\mathcal{RR}$ formulation* (3.1)–(3.4)*, is equal to the optimal objective function value of the $\mathcal{RR}_{Imp}$ formulation* (3.1)–(3.3)*,* (4.1).

Proof. Let a feasible solution to the $\mathcal{RR}_{Imp}$ formulation be represented by a vector $(\bar{x}'', \bar{y}'')$ where $\bar{x}'' = (x''_1, \dots, x''_n)$ and $\bar{y}'' = (y''_1, \dots, y''_n)$, $n = |V|$. Given that, let a vector $(\bar{x}', \bar{y}')$ $(\bar{x}' = (x'_1, \dots, x'_n)$, $\bar{y}' = (y'_1, \dots, y'_n))$ of variables x'_i and y'_i be defined such that $y'_i = y''_i$ for each $i \in V$ and

$$x'_i = \begin{cases} 0, & x''_i \in [0, 1) \\ 1, & x''_i \in [1, +\infty). \end{cases}$$

By this definition, the variables x'_i and y'_i have binary values and therefore satisfy conditions (3.4). Combining the definitions of variables x''_i and binary notation of variables y''_i, it follows that $y'_i = y''_i \leqslant x''_i$. Now, if relations $y''_i = 1$ stand, then inequalities $1 \leqslant x''_i$ imply that $x'_i = 1 \geqslant 1 = y'_i$. More, relations $y''_i = 0$ provide inequalities $0 \leqslant x''_i$, which imply that $x'_i \in \{0, 1\} \geqslant 0 = y'_i$ for each $i \in V$. Therefore, conditions (3.3) are satisfied.
Assuming that $x''_i + \sum_{j \in N_i} y''_j \geqslant 1$, two cases arise:

1) $(\exists j \in N_i)$ such that $y''_j = 1$,
2) $(\forall j \in N_i)$ $y''_j = 0$.

The first case implies that there exists j such that $y'_j = 1$, i.e., $\sum_{j \in N_i} y'_j \geqslant 1$. Therefore, $x'_i + \sum_{j \in N_i} y'_j \geqslant 1$. The second case implies that $1 \leqslant x''_i + \sum_{j \in N_i} y''_j = x''_i$. Now, because of $x''_i \geqslant 1$, it follows that $x'_i = 1$. Given that, conditions (3.2) are satisfied.

Finally, the objective function value of the $\mathcal{RR}$ formulation can be calculated as $\sum_{i \in V} x'_i + \sum_{i \in V} y'_i$, but because of the relations $y'_i = y''_i$ and $x'_i \leqslant x''_i$ it is easy to notice that $\mathrm{Obj}_{\mathcal{RR}} \leqslant \mathrm{Obj}_{\mathcal{RR}_{Imp}}$.

The objective function value of every relaxed minimization problem is less than or equal to the objective function value of the associated original problem, i.e. $\mathrm{Obj}_{\mathcal{RR}_{Imp}} \leqslant \mathrm{Obj}_{\mathcal{RR}}$. Finally, combining the given inequalities the theorem is proven, i.e., $\mathrm{Obj}_{\mathcal{RR}} = \mathrm{Obj}_{\mathcal{RR}_{Imp}}$. □

From the above it can be noted that $2|V|$ binary variables of the existing $\mathcal{RR}$ formulation can be replaced with $|V|$ binary and $|V|$ real variables without losing generality.

4.2. New improved $\mathcal{BVV}$ formulations. Considering the $\mathcal{BVV}$ formulation it can be noted that conditions (3.7) can be omitted. Formulation (3.5), (3.6) and (3.8) will be marked as $\mathcal{BVV}_{\mathrm{Imp1}}$.

Moreover, following the idea for improving the $\mathcal{RR}$ formulation, the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation can be further improved by relaxing binary variables x_i to be real. Such an improvement, (3.5), (3.6) and (4.1), will be marked as $\mathcal{BVV}_{\mathrm{Imp2}}$.

THEOREM 4.2. *Optimal objective function value of the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation* (3.5), (3.6) *and* (3.8) *is equal to the optimal objective function value of the $\mathcal{BVV}$ formulation* (3.5)–(3.8).

PROOF. Let a feasible solution to the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation be represented by a vector $(\bar{x}'', \bar{y}'')$ where $\bar{x}'' = (x_1'', \ldots, x_n'')$, $\bar{y}'' = (y_1'', \ldots, y_n'')$, $n = |V|$. Now, let us define a vector $(\bar{x}', \bar{y}')$ of variables $\bar{x}' = (x_1', \ldots, x_n')$, $\bar{y}' = (y_1', \ldots, y_n')$ such that $y_i' = y_i''$ for each $i \in V$. Given that, we can define two disjunctive sets V_1 and V_2 such that $V_1 \cup V_2 = V$:

1) $x_i'' = 0$ or $y_i'' = 0$, for each vertex $i \in V_1$,
2) $x_i'' = 1$ and $y_i'' = 1$, for each vertex $i \in V_2$.

For each $i \in V_1$, let us define x_i' such that $x_i' = x_i''$. By the definition of the variables x_i'' and y_i'' and because $(x_i', y_i') = (x_i'', y_i'')$, conditions (3.6) and (3.8) are satisfied. Further, for a given set V_1 and because of the binary notations of the variables x_i'' and y_i'' conditions (3.7) are also satisfied:

$$x_i' + y_i' = x_i'' + y_i'' \leqslant \max\{x_i'', y_i''\} \in \{0,1\} \leqslant 1.$$

Now, let us define x_i' for each $i \in V_2$. Because of the definition of the variables y_i' and set V_2 ($y_i' = y_i''$ and $y_i'' = 1$), variables x_i' will be set to be equal to zero, i.e. $x_i' = 0$.

Conditions (3.6)–(3.8) are satisfied again:

$$x_i' + y_i' + \sum_{j \in N_i} y_i' = 0 + 1 + \sum_{j \in N_i} y_i'' = 1 + |V_2| \geqslant 1, \qquad (\forall i) \in V_2$$

$$x_i' + y_i' = 0 + 1 \leqslant 1, \qquad (\forall i) \in V_2$$

$$x_i' = 0 \in \{0,1\}, \qquad y_i' = 1 \in \{0,1\}$$

Therefore, a feasible solution to the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation is a feasible solution to the $\mathcal{BVV}$ formulation. The objective function value can be calculated as follows:

$$\begin{aligned}
\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} &= \sum_{i \in V} x_i'' + 2\sum_{i \in V} y_i'' = \sum_{i \in V_1} x_i'' + 2\sum_{i \in V_1} y_i'' + \sum_{i \in V_2} x_i'' + 2\sum_{i \in V_2} y_i'' \\
&= \sum_{i \in V_1} x_i'' + 2\sum_{i \in V_1} y_i'' + 3|V_2|.
\end{aligned}$$

$$\begin{aligned}
\mathrm{Obj}_{\mathcal{BVV}} &= \sum_{i \in V} x_i' + 2\sum_{i \in V} y_i' = \sum_{i \in V_1} x_i' + 2\sum_{i \in V_1} y_i' + \sum_{i \in V_2} x_i' + 2\sum_{i \in V_2} y_i' \\
&= \sum_{i \in V_1} x_i'' + 2\sum_{i \in V_1} y_i'' + 2|V_2|.
\end{aligned}$$

It is easy to notice that $\mathrm{Obj}_{\mathcal{BVV}} \leqslant \mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}}$.

Now, let us assume that U is a solution set to the $\mathcal{BVV}$ formulation. Omitting condition (3.7) from the $\mathcal{BVV}$ formulation, the solution to the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation can be marked as U_1. It is obvious that $U \subseteq U_1$ and therefore, a feasible solution to the $\mathcal{BVV}$ formulation is also a feasible solution to the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation. By the definition of the global and local minimums it implies that

$$\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} = \min_{U_1} \big(\sum_{i\in V} x_i + 2 \sum_{i \in V} y_i \big) \leqslant \min_{U} \big(\sum_{i\in V} x_i + 2 \sum_{i \in V} y_i \big) = \mathrm{Obj}_{\mathcal{BVV}}$$

Finally, combining $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} \geqslant \mathrm{Obj}_{\mathcal{BVV}}$ and $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} \leqslant \mathrm{Obj}_{\mathcal{BVV}}$ it implies that $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} = \mathrm{Obj}_{\mathcal{BVV}}$. □

THEOREM 4.3. *The optimal objective function value of the $\mathcal{BVV}_{\mathrm{Imp2}}$ formulation* (3.5)*,* (3.6) *and* (4.1) *is equal to the optimal objective function value of the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation* (3.5)*,* (3.6) *and* (3.8)*.*

PROOF. Let a feasible solution to the $\mathcal{BVV}_{\mathrm{Imp2}}$ formulation be represented by a vector $(\bar{x}'', \bar{y}'')$, $\bar{x}'' = (x''_1, \dots, x''_n)$, $\bar{y}'' = (y''_1, \dots, y''_n)$, $n = |V|$. Again, let a vector $(\bar{x}', \bar{y}')$ of variables x'_i ($\bar{x}' = (x'_1, \dots, x'_n)$) and y'_i ($\bar{y}' = (y'_1, \dots, y'_n)$) be defined such that $y'_i = y''_i$ for each $i \in V$ and

$$x'_i = \begin{cases} 0, & x''_i \in [0,1) \\ 1, & x''_i \in [1, +\infty) \end{cases}.$$

The variables x'_i and y'_i have binary values by this definition, therefore they satisfy conditions (3.8).

Assuming that $x''_i + y''_i + \sum_{j \in N_i} y''_j \geqslant 1$, two cases arise:

1) $(y''_i = 1) \vee (\exists j \in N_i)(y''_j = 1)$,
2) $(y''_i = 0) \wedge (\forall j \in N_i)(y''_j = 0)$.

The first case implies that $y''_i = 1$ or there exists j such that $y''_j = 1$, i.e. $y''_i + \sum_{j \in N_i} y''_j \geqslant 1$. Now, knowing that $y'_i = y''_i$ and $x'_i \geqslant 0$ for each $i \in V$, it follows that $x'_i + y'_i + \sum_{j \in N_i} y'_j \geqslant 1$. The second case implies that $1 \leqslant x''_i + y''_i + \sum_{j \in N_i} y''_j = x''_i$. Now, because $x''_i \geqslant 1$ it follows that $x'_i = 1$. Further, because $y_i \in \{0,1\}$ it follows that $x'_i + y'_i + \sum_{j \in N_i} y'_j \geqslant 1$ meaning that conditions (3.2) are satisfied also.

Now, the objective function value of the $\mathcal{BVV}_{\mathrm{Imp1}}$ formulation can be calculated as $\sum_{i \in V} x'_i + 2 \sum_{i \in V} y'_i$. Because of the relations $y'_i = y''_i$ and $x'_i \leqslant x''_i$ it is easy to notice that $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} \leqslant \mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp2}}}$.

Again, knowing that the objective function value of every relaxed minimization problem is less than or equal to the objective function value of the associated original problem, it follows that $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp2}}} \leqslant \mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}}$. Combining the given inequalities the theorem is proven, i.e. $\mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp1}}} = \mathrm{Obj}_{\mathcal{BVV}_{\mathrm{Imp2}}}$. □

Concerning $\mathcal{BVV}_{\mathrm{Imp1}}$ and $\mathcal{BVV}_{\mathrm{Imp2}}$ formulations, the number of constraints is reduced to $|V|$. Further more, by using $\mathcal{BVV}_{\mathrm{Imp2}}$ formulation $2|V|$ binary variables are substituted by $|V|$ binary and $|V|$ real variables.

5. Conclusions

This paper is devoted to the Roman domination problem. For the $\mathcal{RR}$ mathematical formulation it was proven that from a total of $2|V|$ variables, $|V|$ variables can be relaxed to be real. For the $\mathcal{BVV}$ mathematical formulation it was proven that a set of $|V|$ constraints can be excluded. Further more, the number of $2|V|$ binary variables of $\mathcal{BVV}$ mathematical formulation can be also relaxed to $|V|$ binary and $|V|$ real variables. While Theorems 4.1–4.3 shows that improved formulations $\mathcal{RR}_{Imp}$, $\mathcal{BVV}_{\text{Imp1}}$ and $\mathcal{BVV}_{\text{Imp2}}$ are equivalent to the existing ones, there are significant improvements of the computational efforts for solving these formulations.

Operating with lesser number of constraints and integer variables should provide a memory savings in solving the Roman domination problem on large size instances. For instance, on Intel® Core™ i7-4700MQ CPU @ 2.40GHz 2.39GHz with 8GB RAM under Windows 8.1 operating system and based on the new formulations, standard optimization solver CPLEX was able to find the optimal solution value on classes of graphs such as grid, net and planar up to 600 vertices.

Designing an exact method using the proposed MILP formulations is a matter of the future research. Also, in the future work it will be interesting to consider some variants of the Roman domination problem.

References

1. A. Bouchou, M. Blidia, *Criticality indices of roman domination of paths and cycles*, Australas. J. Comb. **56** (2013), 103–112.
2. A. P. Burger, A. P. de Villiers, J. H. van Vuuren, *A binary programming approach towards achieving effective graph protection*, Proc. 2013 ORSSA Annual Conf., ORSSA, 2013, pp. 19–30.
3. V. Currò, *The Roman Domination Problem on Grid Graphs*, Ph.D. thesis, Università di Catania, 2014.
4. P. A. Dreyer, Jr., *Applications and Variations of Domination in Graphs*, Tech. report, 2000.
5. A. Klobučar, I. Puljić, *Some results for roman domination number on cardinal product of paths and cycles*, Kragujevac J. Math. **38**(1) (2014), 83–94.
6. C. H. Liu, G. J. Chang, *Roman domination on strongly chordal graphs*, J. Comb. Optim. **26**(3) (2013), 608–619.
7. P. Pavlič, J. Žerovnik, *Roman domination number of the cartesian products of paths and cycles*, Electron. J. Comb. **19**(3) (2012), p. 19.
8. C. S. ReVelle, K. E. Rosing, *Defendens imperium romanum: a classical problem in military strategy*, Am. Math. Mon. **107**(7) (2000), 585–594.
9. W. Shang, X. Hu, *The roman domination problem in unit disk graphs*; in: *Computational Science – Iccs 2007: 7th Internat. Conf., Beijing China, May 27–30, 2007, Proc., Part III*, Lect. Notes Comput. Sci. 4489, Springer, 2007, pp. 305–312.
10. I. Stewart, *Defend the roman empire!*, Scientific American **281** (1999), 136–138.

Faculty of Mathematics
Department for Applied Mathematics
University of Belgrade
Belgrade
Serbia
marijai@math.rs

(Received 25 03 2015)
(Revised 14 10 2015)