A logo of a globe

Description automatically generatedA blue circle with white text and a logo

Description automatically generatedUNIVERZITET U BANJOJ LUCI

PRIRODNO-MATEMATIČKI FAKULTET

MATEMATIKA I INFORMATIKA, INFORMATIKA

KRIPTOGRAFIJA I RAČUNARSKA ZAŠTITA

***METRIC DIMENSION PROBLEM***

Seminarski rad

Student: Predmetni profesor:

Nikola Đajić (7/19) doc. dr Marko Đukanović

Banja Luka, maj 2025

SADRZAJ

# **UVOD**

Grafovi predstavljaju jedan od osnovnih modela za analizu i reprezentaciju kompleksnih sistema u raznim oblastima, uključujući računarstvo, biologiju, društvene mreže i telekomunikacije. Jedan od problema teorije grafova jeste problem metričke dimenzije (eng. *Metric Dimension Problem – MDP*), koji se odnosi na određivanje minimalnog skupa čvorova, koje nazivamo rješavajućim skupom, tako da se svi preostali čvorovi u grafu mogu jednoznačno identifikovati na osnovu svoje udaljenosti do čvorova tog skupa. Ovaj problem jeste NP-težak, što znači da ne postoji efikasan algoritam koji ga može riješiti za sve instance u polinomijalnom vremenu. Zbog toga se koriste različite heurističke i optimizacione metode, kako bi se pronašla optimalna ili približna rješenja.

U ovom radu implementirani su genetski algoritam (eng. *Genetic Algorithm – GA*) i algoritam optimizacije kolonijama mrava (eng. *Ant Colony Optimization – ACO*). Dobijeni rezultati su upoređeni sa algoritmima varijabilna pretraga susjedstva (eng. *Variable Neighborhood Search – VNS*) i cjelobrojno linearno programiranje (*Integer Linear Programming – ILP*). Izvorni kod dostupan je putem javnog GitHub repozitorijuma, na linku: <https://github.com/NikolaDjajic/MDP>

U nastavku rada predstavljena je formalna definicija problema, pregled relevantne literature, opis primijenjenih algoritama, kao i dobijeni rezultati.

# **PROBLEM METRIČKE DIMENZIJE GRAFOVA**

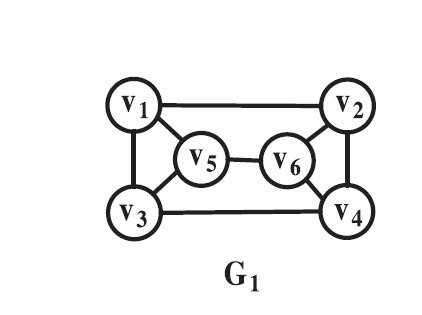
Metrička dimenzija grafova prvi put je formalno uvedena u radu Harary i Melter (1976), a kasnije su je detaljno analizirali Chartrand i saradnici 2000. godine. Problem metričke dimenzije predstavlja značajan aspekt teorije grafova, sa primjenama u različitim oblastima kao što su navigacija robota, lociranje grešaka u mrežama, identifikacija čvorova u komunikacionim sistemima i bioinformatika.

Neka je G=(V, E) neprazan, cikličan i povezan graf. Za skup čvorova W ⊆ V, kažemo da W „rješava“ graf G ako za svaki par različitih čvorova u, v ∈ V postoji čvor w, takav da w ∈ W i rastojanje d(u, w) ≠ d(v, w). Drugim riječima, svaka dva čvora u grafu se mogu razlikovati na osnovu svojih udaljenosti do bar jednog čvora iz skupa W.

Za dati skup čvorova S={s1,s2,...,sk}, vektor rastojanja za čvor v definiše se kao:

r(v,S)=(d(v,s1),d(v,s2),...,d(v,sk)),

pri čemu je d(v,si) najkraće rastojanje između čvorova v i si​. Skup S je skup rješenja (eng. *Resolving Set*) ako su svi vektori r(v,S) jedinstveni za svako v∈V∖S. Najmanji takav skup označava se kao minimalni rješavajući skup, a njegova kardinalnost predstavlja metričku dimenziju grafa G, označenu kao β(G).



*Slika 1. Primjer grafa G1*

Na Slici 1, prikazan je graf G1, nad kojim će biti prikazan problem metričke dimenzije.Graf G1 posjeduje 6 čvorova i 9 ivica. Rastojanje između dva povezana čvora je uvijek jednako jedan.

Skup S1 = {v1,v2,v3} je jedan od rješenja, zato što:

* r(v1, S1) = **(0, 1, 1)**;
* r(v2, S1) = **(1, 0, 2)**;
* r(v3, S1) = **(1, 2, 0)**;
* r(v4, S1) = **(2, 1, 1)**;
* r(v5, S1) = **(1, 2, 1)**;
* r(v6, S1) = **(2, 1, 2)**.

r(vi , S) je rastojanje od čvora vi do svakog od čvorova iz skupa S, te je isto što i: r(vi, {v1,v2,v3}).

Kao što se vidi, ne postoje dva identična skupa što znači da pomoću S1 možemo jednoznačno identifikovati svaki čvor koji nije u S1 na osnovu njegove udaljenosti od čvorova S1. Skup S1 sadrži 3 čvora pa je i kardinalnost skupa jednaka 3.

Međutim skup S1 nije minimalni rješavajući skup, zato što postoji i S2 = {v1,v3} koji je takođe skup rješenja.

* r(v1, S2) = **(0, 1)**;
* r(v2, S2) = **(1, 2)**;
* r(v3, S2) = **(1, 0)**;
* r(v4, S2) = **(2, 1)**;
* r(v5, S2) = **(1, 1)**;
* r(v6, S2) = **(2, 2)**.

S2 predstavlja bolje rješenje u odnosu na S1, zato što je kardinalnost skupa jednaka 2. Takođe S2 nije jedinstveno minimalno rješenje pošto postoji i skup S3 = {v2,v4} kojem je kardinalnost jednaka isto 2 kao i skupu S2.

# **PREGLED LITERATURE**

*Tabela 1. Pregled relevantne literature*

|  |  |
| --- | --- |
| Harary i Melter [1] | Prvi put formalno uvedena metrička dimenzija. |
| Chartrand i saradnici [2] | Detaljnija analiza metričke dimenzije. |
| Kratica i saradnici [3] | Predložen je prvi metaheuristički pristup problemu metričke dimenzije grafova. Genetski algoritam predstavljen u ovom radu koristi binarno kodiranje i standardne genetske operatore prilagođene problemu. |
| Chartand i saradnici [2];  Currie i Oellermann [4] | Formulacije problema metričke dimenzije pomoću ILP-a |
| Kratica i saradnici [5] | ILP formulacija za problem minimalnog dvostrukog skupa razrješana. |
| Peters-Fransen i Oellermann [6];  Cáceres i saradnici [7] | Bavili su se određivanjem donjih i gornjih granica za metričku dimenziju Karatezijanovog produkta grafova. |
| Yero i saradnici [8] | Istražili metričku dimenziju korona produkta grafova, te dali teorijske granice |
| Hernando i saradnici [9] [10] | Istraživali vezu između metričke dimenzije i drugih grafičkih invarijanti, kao što su prečnik, broj čvorova i stepen čvorova. |
| Chappell i saradnici [11] | Uspostavili relaciju između metričke i particione dimentije grafa. |
| Bailey i Meagher [12] | Analizirali su metričku dimenziju Grassmannovih grafova. |
| Bailey i Cameron [13] | Ispitivali metričke dimenzije Johnsonovih i Knezerovih grafova. |
| Javaid i saradnici [14];  Husnine i Kousar [15] | Istraživali metričke dimentije generalizovanih Petersonovih grafova. |
| Cáceres i saradnici [16];  Rebatel i Thiel [17] | Pokazali da neki beskonačni grafovi imaju beskonačnu metričku dimenziju. |
| Beerliova i saradnici [18] | Primijenili su MDP u domenu telekomunikacija. |
| Chartrand i saradnici [2] | Predložili su primjenu MDP u hemiji, za identifikaciju jedinstvenih pozicija u molekulskim strukturama. |
| Khuller i saradnici [19] | Primijenili su MDP u navigaciji robota. |

# **GENETSKI ALGORITAM**

Genetski algoritam pripada heurističkim metodama (metode koje ne garantuju optimalno rješenje). Inspiracija za ovaj algoritam leži u Teoriji evolucije Čarlsa Darvina.

Neki od pojmova koje bi trebalo da definišemo kada pričamo o genetskom algoritmu:

* Jedinka = svaka jedinka je moguće rješenje problema.
* Populacija = skup jedinki koji predstavlja podskup prostora pretrage.
* Prilagođenost (eng. fitness) = ocjena kvaliteta date jedinke.
* Genetski kod = način na koji se predstavlja jedna jedinka.
* Geni = dijelovi genetskog koda.

Takođe, postoje i genetski operatori, a to su:

*Selekcija*

Ovim procesom se biraju jedinke nad kojima će se poslije vršiti ukrštanje i mutacija. Jedinke se biraju na osnovu individualne prilagođenosti, one sa većom prilagođenošću su u prednosti da budu odabrane za nastavak. Neke od načina selekcije su ruletska, turnirska, eliminaciona i druge.

*Ukrštanje*

Slično kao i u samoj evoluciji, ukrštanje predstavlja miješanje gena jedinki. Kao rezultat ove operacije dobijamo novu jedinku koja nosi potencijalno dobre gene svojih roditelja.

*Mutacija*

Ovaj proces predstavlja mijenjanje gena. Time se omogućava vraćanje dobrih gena koji su se izgubili prilikom prethodnih radnji. Posljedica izvršavanja prethodno definisanih radnji jeste nastanak nove generacije.

Naša implementacija genetskog algoritma se malo razlikovala od klasične pa je možemo zvati i izmjenjeni genetski algoritam. Početnu populaciju biramo nasumično (engl. *random*), tako da svaki čvor ima 50% šanse da se nađe u početnom skupu.

Za selekciju smo koristili težinsku selekciju, to je vrsta selekcije koja bira jedinke (rješenja) za sljedeću generaciju na osnovu njihove *fitness* vrijednosti. Ukrštanje smo radili obično, tako što presječemo dvije jedinke i onda ih spojimo sa polovinama druge jedinke.

Mutaciju smo malo izmjenili tako da ukoliko gen mutira on može samo da mutira na bolje, to jeste da taj čvor koji je bio u datoj jedinki ispadne iz nje. Ovim pristupom neće se moći vratiti neki gen ukoliko je odbačen, ali će se jako brzo smanjivati broj čvorova u jednikama što je nama i potrebno, pošto krećemo sa jako lošim početnim rješenjem. Ovakav način mutacije dao je neuporedivo bolje rezultate i brže vrijeme izvršavanja algoritma.

Pomoći *fitness* funkcije procjenjujemo kvalitet datog rješenja. To radimo tako što provjeravamo da li postoje dva čvora u grafu koji imaju identična rastojanja do svih čvorova iz rješavajućeg skupa. Ukoliko ne postoje vraćamo dužinu tog rješenja kao *fitness* vrijednost, međutim ako postoji onda isto tako vraćamo dužinu rješenja + 100 \* broj nerješenih parova. Time dajemo jako veliku *fitness* vrijednost rješenjima koja ne predstavljaju rješavajući skup.

Sve najkraće puteve između čvorova smo unaprijed izračunali kako bismo smanjili vrijeme izvršavanja algoritma.

# **ACO – OPTIMIZACIJA KOLONIJE MRAVA**

ACO je heuristički optimizacioni algoritam inspirisan ponašanjem stvarnih mrava u prirodi, posebno njihovom sposobnošću da pronalaze najkraći put između izvora hrane i mravinjaka. Algoritam je prvi put predstavio Marco Dorigo 1992. godine u okviru svoje doktorske disertacije.

Osnovne ideje i pojmovi:

* Mrav = predstavlja pojedinačno rješenje problema;
* Kolonija = skup svih mrava (rješenja) koji se kreću kroz prostor pretrage;
* Feromoni = vještačka “hemijska” oznaka koja se koristi za memorisanje prethodnih dobrih rješenja. Mravi više preferiraju puteve s većom koncentracijom feromona, i
* Heuristička informacija = dodatna informacija o kvalitetu rješenja, npr. inverzna udaljenost kod problema najkraće putanje.

Glavni procesi u ACO algoritmu:

*Kretanje mrava kroz prostor rješenja*

Svaki mrav gradi rješenje korak po korak, birajući naredni korak na osnovu kombinacije količine feromona i heurističke vrijednosti (npr. dužine puta). Ova selekcija je probabilistička, što znači da bolje opcije imaju veću šansu da budu izabrane, ali i lošije se povremeno istražuju – radi očuvanja raznolikosti.

*Ažuriranje feromona*

Nakon što svi mravi završe konstruisanje svojih rješenja, feromoni se ažuriraju. Dobri putevi (bolja rješenja) dobijaju više feromona, dok stari feromoni postepeno isparavaju. Ova ravnoteža između ojačavanja dobrih puteva i isparavanja loših sprečava preuranjenu konvergenciju.

*Isparavanje feromona*

Ova operacija smanjuje vrijednosti feromona na svim putevima kako bi se stari, neefikasni putevi vremenom zaboravili. Time se podstiče istraživanje novih rješenja.

Na početku svakom čvoru u grafu dodjeljujemo istu količinu feromona. Pored toga izračunavamo heurističku vrijednost svakog čvora kao njegova betweenness centrality – što je čvor “važniji” za povezivanje ostalih, to mu heuristika daje veću privlačnost. Postavljamo parametre tako da feromon i heuristika imaju različitu važnost (alpha i beta), određujemo koliko brzo feromoni isparavaju (evaporation), te ukupnu količinu feromona koja se dodjeljuje uspješnim rješenjima (Q). Takođe biramo broj mrava i broj iteracija koje će simulirati. Vrijednosti svih ovih parametara zavise od datog grafa.

Svaki mrav gradi svoje rješenje iterativnim izborom čvorova: u svakom koraku posmatra preostale čvorove i bira ih na osnovu kombinovane ocjene feromona i heuristike. Čvorovi s jačim feromonskim tragom i višom heurističkom vrijednošću imaju veću vjerovatnoću da budu odabrani.

Kvalitet svakog rješenja procjenjujemo pomoću iste logike kao u genetskom algoritmu: mjeri se veličina odabranog skupa, uz snažnu kaznu za svaki par čvorova koji ostane nerješiv. Time osiguravamo da se prvo traže bolja rješenja, a potom smanjujemo njihovu veličinu.

Nakon što svi mravi naprave svoja rješenja u jednoj iteraciji, traži se najbolji među njima. Zatim isparava dio feromona sa svih čvorova kako bismo održali raznovrsnost pretrage, i na čvorove iz najboljeg rješenja dodajemo novu dozu feromona – ta doza je veća ako je rješenje bolje.

Kao i kod genetskog algoritma sve najkraće puteve između čvorova smo unaprijed izračunali.

# **REZULTATI**

*Tabela 2. Eksperimentalni rezultati*

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Grafovi | | | | VNS | | GA | | ACO | | PULP1 | | PULP2 | |
| Naziv | |V| | |E| | *davg* | β(G) | t | β(G) | t | β(G) | t | β(G) | t | β(G) | t |
| Graf-10 | 10 | 18 | 1,8 | **4** | 0.06 | **4** | 0.1 | / | / | **4** | 0.04 | **4** | 0.2 |
| Graf-11 | 11 | 20 | 1,81 | **4** | 0.04 | **4** | 0.04 | / | / | **4** | 0.04 | **4** | 0.3 |
| Graf-12 | 12 | 40 | 3,3 | **3** | 0.07 | **3** | 0.04 | **3** | 0.09 | **3** | 0.06 | **3** | 0.4 |
| Graf-14 | 14 | 44 | 3 | **3** | 0.06 | **3** | 0.06 | **3** | 0.06 | **3** | 0.05 | **3** | 0.7 |
| Graf-17 | 17 | 38 | 2,2 | **2** | 0.15 | **2** | 0.08 | **2** | 0.06 | **2** | 0.07 | **2** | 0.8 |
| *johnson8-2-4* | 28 | 420 | 15 | **6** | 0.3 | **6** | 0.27 | **6** | 0.44 | **6** | 6.6 | **6** | 3.1 |
| *MANN-a9* | 45 | 918 | 40 | **16** | 1.53 | 17 | 16 | **16** | 5.2 | **16** | 21.9 | **16** | 88 |
| *hamming6-4* | 64 | 704 | 22 | **7** | 6 | 8 | 38.4 | 9 | 14 | / | 600 | / | 600 |
| *hamming6-2* | 64 | 2000 | 57 | **17** | 10.2 | 20 | 24 | 19 | 14.5 | / | 600 | / | 600 |
| *johnson8-4-4* | 70 | 2000 | 53 | **10** | 5.56 | 11 | 13.5 | 12 | 17.3 | / | 600 | / | 600 |
| *keller4* | 171 | 9000 | 110 | **12** | 13.4 | 16 | 52.6 | 14 | 60.5 | / | 600 | / | 600 |
| *hamming8-4* | 256 | 21000 | 164 | **13** | 24.7 | 18 | 168.4 | 16 | 167.2 | / | 600 | / | 600 |
| Csp50 | 53 | 176 | 6.6 | **8** | 1.5 | 9 | 4.5 | 45 | 5.2 | **8** | 1.5 | **8** | 53.1 |
| Csp100 | 106 | 722 | 13.6 | **11** | 6.3 | 14 | 26.4 | 17 | 58.5 | **10** | 592 | / | 600 |
| Csp150 | 401 | 1505 | 7.5 | **13** | 24.1 | 17 | 98.5 | 19 | 59.3 | / | 600 | / | 600 |
| Gcol1 | 100 | 2487 | 49.7 | **9** | 5.4 | **9** | 28.1 | 10 | 9.1 | / | 600 | / | 600 |
| Gcol5 | 100 | 2450 | 49 | **9** | 5.3 | 10 | 27.2 | 10 | 9.1 | / | 600 | / | 600 |
| Gcol11 | 100 | 2467 | 49.3 | **9** | 5.6 | 10 | 27 | 10 | 13.4 | / | 600 | / | 600 |
| Gcol15 | 100 | 2528 | 50.5 | **9** | 6 | **9** | 26.6 | 10 | 8.9 | / | 600 | / | 600 |
| Gcol18 | 100 | 2472 | 49.4 | **9** | 6.3 | 10 | 23.2 | 10 | 9.1 | / | 600 | / | 600 |
| Gcol20 | 100 | 2420 | 48.4 | **9** | 5.5 | 10 | 27.2 | 10 | 9.4 | / | 600 | / | 600 |
| Gcol23 | 300 | 22393 | 149.2 | **12** | 41.6 | 15 | 156 | 13 | 81 | / | 600 | / | 600 |
| Gcol24 | 300 | 22446 | 149.6 | **12** | 63 | 14 | 214 | 13 | 88 | / | 600 | / | 600 |
| Gcol26 | 300 | 22601 | 150.6 | **12** | 54.3 | 14 | 226 | 13 | 63 | / | 600 | / | 600 |
| Gcol30 | 300 | 22543 | 150.2 | **12** | 92 | 14 | 235 | 13 | 46 | / | 600 | / | 600 |

Tabela sadrži sljedeće podatke:

* **|V|** – broj čvorova,
* **|E|** – broj grana,
* **davg** – prosječan stepen čvora,
* **β(G)** – metrička vrijednost grafa(vrijednost koju tražimo),
* **t** – vrijeme izvršavanja algoritma u sekundama.

U Tabeli 2. su prikazani rezultati pet algoritama – VNS, GA, ACO, PULP1 i PULP2 – primijenjenih na različite grafove. Algoritmi GA (genetski algoritam) i ACO (algoritam mravlje kolonije) su implementirani u ovom radu, dok su algoritmi VNS, PULP1 i PULP2 implementirani na osnovu ideja iz rada

## POREĐENJE ALGORITAMA

***Manje instance***

Na manjim grafovima svi algoritmi postižu odlične rezultate. GA i ACO se pokazuju kao konkurentni sa VNS, PULP1 i PULP2, ostvarujući iste vrijednosti ciljne funkcije, uz slično vrijeme izvršavanja. To ukazuje da su implementacije ispravne i daju kvalitetna rješenja na jednostavnijim instancama. ACO nije uspio da nađe rješenja u jako malim grafovima zbog načina na koji algoritam radi. Na istanci Csp50 se takođe vidi nedostatak ACO algoritma u radu sa manjim grafovima.

***Srednje instance***

Kod srednje velikih instanci, GA i ACO ostvaruju rješenja malo lošija nego VNS, uz duže vrijeme izvršavanja. ACO pokazuje nešto slabije rezultate od GA na nekim instancama (npr. hamming6-4). Kod instanci kao što su Csp100 i Csp150, GA se pokazuje kao uspješniji u poređenju s ACO, dok PULP metode nisu uspjele da pronađu rješenje u zadatom vremenskom ograničenju od 10 minuta.

***Velike instance***

Na velikim instancama, GA i ACO uspjevaju da pronađu rješenja, ali su ona malo lošijeg kvaliteta u odnosu na VNS. GA uzima puno više vremena u odnosu na VNS i ACO. ACO i VNS imaju slično vrijeme izvršavanja, ali ACO daje malo lošije rezultate.

Važno je dodati da VNS algoritam, koji smo implementirali na osnovu rada, posjeduje nedostatak koji se ogleda u tome da u nekim slučajevima ne dobijemo skup koji je rješavajući ili najbolje rješenje za taj graf. Razlog tome je način na koji je napisan, to jeste ovaj algoritam ne čuva najbolje rješenje već na kraju zadatog broja iteracija vraća posljednje rješenje. Tu ponekad dolazi do greške, te postoji mogućnost da dobijeni skup ne predstavlja dobijeno rješenje, već rješenje generisano u određenom vremenskom intervalu.

Oba PULP algoritma su davali najbolje rezultate, dok su mogli da pronađu rješenje u roku od 10 minuta, kao što vidimo u tabeli to se desilo za samo nekoliko manjih grafova. Takođe PULP1 se pokazao dosta brži od PULP2 algoritma.

ACO i GA su se pokazali dosta slični što se tiče rezultata, međutim ACO je dosta brži u odnosu na GA. Glavni razlog je što ACO počinje sa dosta dobrim početnim skupom i onda mu je potrebno manje vremena da usavrši taj rezultat. Dok GA počinje sa poprilično lošim početnim skupom (nasumična selekcija), i samim tim mu je potrebno dosta više vremena da dobije bolji rezultat.

# **ZAKLJUČAK**

U ovom radu istražen je NP‑težak problem metričke dimenzije grafova (MDP), čiji je cilj pronaći najmanji rezolventni skup čvorova tako da se svaki par čvorova može jednoznačno razlikovati prema udaljenosti do članova tog skupa.

Izmijenjeni GA pokazao se robusnim u pronalaženju malih rezolventnih skupova zahvaljujući specifičnoj mutaciji „samo na bolje“ i težinskoj selekciji. Time se brzo smanjuje veličina početno loših rješenja.

ACO algoritam, naslonjen na feromonske tragove i betweenness heuristiku, imao je znatno kraće vrijeme izvršavanja u odnosu na GA, dok je kvalitet rješenja bio uporediv.

 **Poređenje pet metoda** na skupu standardnih grafova (Graf‑10…Gcol30) pokazalo je da:

* Na malim instancama svi algoritmi dostižu optimalne dimenzije uz zanemarive razlike u vremenu.
* Na srednjim instancama GA blago nadmašuje ACO u kvaliteti, ali ACO je brži. ILP‑metode često ne pronalaze rješenje unutar 600 s.
* Na velikim instancama PULP1/PULP2 daju najbolje dimenzije, ali uz puno duže vrijeme izvršavanja. GA i ACO nude kompromis – brza, približno optimalna rješenja.

# **LITERATURA**

[1] Harary, F., Melter, R., 1976. On the metric dimension of a graph. Ars Combinatoria 2, 191–195.

[2] Chartrand, G., Eroha, L., Johnson, M., Oellermann, O., 2000. Resolvability in graphs and the metric dimension of a graph. Discrete Applied Mathematics 105, 99– 113.

[3] Kratica, J., Kovačević -Vujčić, V., Čangalović, M., 2009. Computing the metric dimension of graphs by genetic algorithms. Computational Optimization and Applications 44, 343–361.

[4] Currie, J., Oellerman, O., 2001. The metric dimension and metric independence of a graph. Journal of Combinatorial Mathematics and Combinatorial Computing 39, 157–167.

[5] Kratica, J., Čangalović, M., Kovačević -Vujčić, V., 2009b. Computing minimal doubly resolving sets of graphs. Computers & Operations Research 36, 2149–2159.

[6] Peters-Fransen, J., Oellermann, O., 2006. The metric dimension of cartesian products of graphs. Utilitas Mathematica 69, 33–41.

[7] Cáceres, J., Hernando, C., Mora, M., Pelayo, I., Puertas, M., Seara, C., Wood, D., 2007. On the metric dimension of cartesian products of graphs. SIAM Journal in Discrete Mathematics 21, 423–441.

[8] Yero, I., Kuziak, D., Rodriguez-Velazquez, J., 2011. On the metric dimension of corona product graphs. Computers and Mathematics with Applications 61, 2793–2798.

[9] Hernando, C., Mora, M., Pelayo, I., Seara, C., Wood, D., 2007. Extremal graph theory for metric dimension and diameter. Electronic Notes in Discrete Mathematics 29, 339–343.

[10]Hernando, C., Mora, M., Pelayo, I., Seara, C., Wood, D., 2010. Extremal graph theory for metric dimension and diameter. Electronic Journal of Combinatorics 17, 1– 28.

[11] Chappell, G., Gimbel, J., Hartman, C., 2008. Bounds on the metric and partition dimensions of a graph. Ars Combinatoria 88, 349–366.

[12] Bailey, R., & Meagher, K., 2011. On the metric dimension of Grassmann graphs. Technical Report. arXiv:1010.4495.

[13] Bailey, R., Cameron, P., 2011. Base size, metric dimension and other invariants of groups and graphs. Bulletin of the London Mathematical Society 43, 209–242.

[14] Javaid, I., Rahim, M., Ali, K., 2008. Families of regular graphs with constant metric dimension. Utilitas Mathematica 75, 21–33.

[15] Husnine, S., Kousar, I., 2010. A subfamily of a generalized petersen graph p(n, 3) with constant metric dimension. Utilitas Mathematica 81, 111–120.

[16] Cáceres, J., Hernando, C., Mora, M., Pelayo, I., Puertas, M., 2009. On the metric dimension of infinite graphs. Electronic Notes in Discrete Mathematics 35, 15– 20.

[17] Rebatel, F., Thiel, E., 2011. Metric bases for polyhedral gauges. Lecture Notes in Computer Science 6607, 116–128.

[18] Beerliova, Z., Eberhard, F., Erlebach, T., Hall, A., Hoffmann, M., Mihalák, M., Ram, L., 2006. Network discovery and verification. IEEE Journal on Selected Areas in Communications 24, 2168–2181.

[19] Khuller, S., Raghavachari, B., Rosenfeld, A., 1996. Landmarks in graphs. Discrete Applied Mathematics 70, 217–229.

[20]