*Abstract*: În această lucrare doresc să prezint rezultatele obţinute prin utilizarea unui algoritm în calcularea unor locaţii pentru spitalele regionale ce ar putea fi construite în Romania. Despre spitalele regionale din România se discută de ceva timp în presa locală, locaţiile finale fiind stabilite, de către cei responsabili, în primele 8 oraşe, luate în funcţie de numărul de locuitori, precum şi un oraş aflat in centrul ţării. În cadrul abordării de faţă utilizăm un algoritm genetic, metodă inspirată din evoluţia naturală, care ne va oferi o serie de rezultate alternative la soluţia propusă de oficiali. Ca şi în varianta oficială, în cadrul metodei abordate, pentru obţinerea rezultatelor vom ţine cont de populaţie, dar nu numai din cadrul localităţilor populate, ci din cadrul unui număr mai mare de localităţi de pe teritoriul ţării. Populaţia acestor localităţi, ponderată cu distanţa dintre ele şi posibilele soluţii vor contribui la calcularea soluţiei finale.

**De la regionalizare la spitale regionale**

Conform Consiliului Europei[[1]](#footnote-1), regionalizarea este procesul de transfer al puterii de la guvernul central în regiuni, pentru o mai bună aplicare a principiului subsidiarității, în cadrul solidarității naționale sau federale. Acesta include înființarea, lărgirea sau împuternicirea autorităților și transferul de competențe și responsabilități către regiuni. Comitetul de guvernanță și Camera regiunilor din Congres sunt responsabile pentru problemele juridice și politice legate de regionalizare în statele membre ale Consiliului Europei. Acestea examinează în special evoluțiile privind organizarea instituțională și administrativă a regiunilor, competențele și autonomia financiară a acestora.

Din punct de vedere statistic, regiunea poate avea două semnificații distincte (se poate referi la o zonă geografică)[[2]](#footnote-2):

* la nivel subnațional, o subdiviziune a unei țări, la diferite niveluri posibile, dar cel mai frecvent la nivelul 1 al Nomenclatorului unităților teritoriale pentru statistici (NUTS);
* la nivel supranațional, o regiune a lumii precum „America Latină”, „Africa Subsahariană”, „Asia de Sud-Est” etc ., sau diferite organizări la nivelul regiunilor mondiale: de ex. AU, ASEAN, CAN, NAFTA, UE.

În cadrul Uniunii Europene, există, în acest moment 240 de euro-regiuni[[3]](#footnote-3), create pe baza NUTS, nomenclator care asigură comparabilitatea acestora la nivel european. Fiind stat membru a Uniunii Europene, în România au fost create 8 regiuni de dezvoltare: Nord-Vest(RO11), Centru (RO12), Nord-Est(RO21), Sud-Est(RO22), Sud-Muntenia (RO31), Bucuresti-Ilfov (RO32), Sud-Vest Oltenia (RO41) şi Vest (RO42). Pornind de la această structură, reprezentanţii Ministerului Sănătăţii din România au lansat ideea construirii a unui spital de dimensiuni superioare amplasat în regiunea RO32 (cea mai dezvoltată de pe teritoriul României, având şi o densitate a populaţiei superioară) şi alte 8 spitale regionale care să deservească populaţia din celelalte regiuni din România. Aceste spitale ar urma să fie amplasate în localităţile Iaşi (RO21), Cluj (RO11), Craiova (RO41), Braşov (RO12), Timişoara (RO42), Târgu-Mureş (RO12), Brăila/Galaţi (RO22), Constanţa (RO22). La o privire mai atentă asupra oraşelor alese se poate observa că 8 din cele nouă spitale se vor construi în primele 8 oraşe, luate din punctul de vedere al numărului de locuitori, singurul oraş ce nu intră în acest top fiind Târgu Mureş. Regionalizarea sănătății a fost adoptată de mai multe țări pentru a îmbunătăți accesul populației la serviciile pentru sănătate[[4]](#footnote-4),[[5]](#footnote-5). Organizația Mondială a Sănătății (OMS) o definește ca distribuirea rațională a serviciilor medicale pe întreg teritoriul, asigurând că serviciile și facilitățile asigură toate nivelurile de îngrijire (primară, secundară și terțiară) cu acces facil la populație și îngrijire rentabilă[[6]](#footnote-6). Totuşi, în anumite cazuri, această decizie poate duce la un sentiment de inegalitate în distribuirea geografică a viitoarelor construcţii, dar şi alte neplăceri cu rezultate directe sau indirecte în multe domenii[[7]](#footnote-7):

* regiunea RO31 nu are atribuit nici un spital, acest lucru fiind datorat amplasării unui spital în regiunea RO32, aceasta din urmă fiind în centrul regiunii RO31, atrăgând în acest mod locuitorii acesteia;
* regiunea cea mai populată (RO21) are atribuit un singur spital, în timp ce alte două regiuni cu numărul de locuitori inferior au câte două spitale;
* se supra-aglomerează oraşele alese, ele având deja unităţi spitaliceşti care să deservească populaţia din localitate;
* ajungem la creşterea decalajelor dintre oraşe din punctul de vedere al unităţilor arondate sistemului sanitar, lucru care ar influenţa într-un mod negativ economia, educaţia, densitatea populaţiei, etc.

În continuare vom prezena metoda aleasă pentru găsirea unor locaţii pentru cele 9 spitale propuse, locaţii care ar duce înlătura neajunsurile prezentate anterior, datele de intrare pentru metoda aleasă, rezultatele obţinute şi compararea lor cu cele propuse la nivel central, dar şi concluziile pe care le-am dedus în urma acestei cercetări.

**Metoda utilizată în căutarea soluţiilor**

In the 1950s and the 1960s several computer scientists independently studied evolutionary systems with the idea that evolution could be used as an optimization tool for engineering problems. The idea in all these systems was to evolve a population of candidate solutions to a given problem, using operators inspired by natural genetic variation and natural selection[[8]](#footnote-8).

Algoritmii genetici sunt o familie de modele de calcul inspirate din evoluție. Aceşti algoritmi codifică o soluție potențială pentru o problemă specifică folosind un cromozom (exprimat, de cele mai multe ori printr-un tablou) și aplică operatori de recombinare asupra acestor structuri (cromozomi), astfel încât să se păstreze informaţiile critice aferente acestora[[9]](#footnote-9).

Holland's 1975 book Adaptation in Natural and Artificial Systems presented the genetic algorithm as an abstraction of biological evolution and gave a theoretical framework for adaptation under the GA. Holland's GA is a method for moving from one population of "chromosomes" to another population by using some natural−inspired operators of crossover, mutation, and inversion. Each chromosome consists of "genes", each gene being an instance of a particular "allele". The selection operator chooses those chromosomes in the population that will be allowed to reproduce, and on average the fitter chromosomes produce more offspring than the less fit ones. Crossover exchanges subparts of two chromosomes; mutation randomly changes the allele values of some locations in the chromosome; and inversion reverses the order of a contiguous section of the chromosome, thus rearranging the order in which genes are arrayed[[10]](#footnote-10).

Pentru studiu de caz prezentat cromozomul va avea 10 gene, primele 9 gene fiind codurile oraşelor din baza de date, cea de-a zecea fiind folosită pentru stocarea valorii unei funcţii obiectiv, valoare care va influenţa modul de selecţie a cromozomilor utilizaţi în evoluţie.

Operatorii pe care i-am utilizat în cadrul acestui algoritm sunt mutaţia şi încrucişarea. Pentru ambii operatori am folosit o strategie elitistă pentru selectarea cromozomilor participanţi[[11]](#footnote-11). Astfel, în cazul mutaţiei, generăm, aleatoriu, un cromozom, dintre cei mai buni existenţi în populaţia curentă, precum şi o genă (de la 0 la 8, valori ce indică poziţia genei în cromozom). Valoarea acelei gene va fi înlocuită cu altă valoare(id de oraş) care nu există în cromozomul curent. În figura 1 avem o reprezentare grafică a acestui proces.

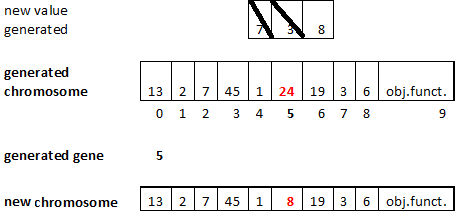


Fig. 1 Mutation

Pentru operatorul de încrucişare, pentru selectarea cromozomilor am aplicat aceeaşi strategie elitistă. După generarea a doi cromozomi şi a unei gene am obţinut un cromozom copil ca-n figura 2.

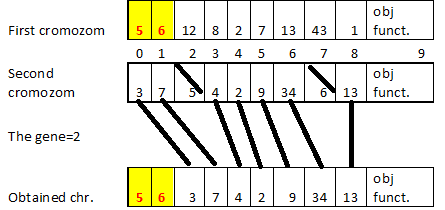


Fig.2 Încrucişare

**Computational data used**

Algoritmul genetic utilizat pentru rezolvarea problemei propuse foloseşte o populaţie de 100 de cromozomi, populaţie care evoluează timp de 100 de generaţii. La fiecare generaţie am aplicat 20 de mutaţii şi 10 încrucişări. Primele 9 gene ale fiecărui cromozom vor fi id-uri de oraşe din România. Numărul de oraşe pe care l-am luat în considerare este de 52, fiind oraşe cu populaţie de peste 40.000 de locuitori[[12]](#footnote-12). Pentru calculul funcţiei obiectiv, cea pe care doresc s-o optimizez, voi utiliza şi distanţele dintre oraşe[[13]](#footnote-13),[[14]](#footnote-14).

Un prim pas în cadrul algoritmului folosit este generarea populaţiei iniţiale. Id-urile care vor reprezenta genele fiecărui cromozom vor fi între 1 şi 52 (1 fiind Bucureşti, oraşul cu populaţia cea mai numeroasă, iar 52 este Mangalia, un oraş cu o populaţie peste 40.000 de locuitori aflat în regiunea RO22). Conform datelor statistice populaţia României în 2016 era de 22.241.718. Populaţia celor 52 de oraşe luate în considerare ajungea la 8.761.689 locuitori, reprezentând peste 39% din populaţia finală.

O atenţie deosebită trebuie s-o acordăm calculării funcţiei obiectiv. Compunerea unei funcţii, prin a cărei optimizare să ajungem la obţinerea de rezultate bune pentru problema studiată, este decisivă. În cazul de faţă la calcularea valorii acestei funcţii utilizez populaţia din oraşe, ponderată cu o valoare ce depinde de distanţa dintre acel oraş şi un oraş aflat în soluţie. Pentru calcularea funcţiei obiectiv am stabilit utilizarea termenului de vecinătate. Astfel, dacă distanţa dintre două oraşe este mai mică sau egală cu 200 km, atunci le considerăm vecine. Cunoscând acest lucru putem spune că valoarea funcţiei obiectiv se calculează cu folosind pseudocodul:

**int** val = 0;  
**for** (**int** i = 0; i < *9*; i++) {  
 **for** (**int** j = 0; j < *id*; j++) {  
 val += pop[i].getDifPondCum(j) \* *loc*(j).getLoc();  
 }  
}  
**return** val;

,where:

-val is the function value;

-loc(j).getLoc()-return number of people from j city;

-pop[i].getDifPondCum(j):

- return 1, if i is equal with j;

-otherwise, return 1-1/(distance between i and j), with the specification that the sum for the values ​​returned for the same city j within a chromosome cannot exceed the value 1 (if it reaches 1, the returned value will be 0).

După generarea populaţiei iniţiale, în fiecare generaţie ce urmează se vor aplica operatorii de încrucişare şi mutaţie. Aceşti doi operatori, împreună cu o abordare elitistă, legat de populaţia care va supravieţui, vor duce la obţinerea de rezultate satisfăcătoare pentru cazul studiat. Ca să observăm câştigul dat de utilizarea acestui algoritm, ar trebui să studiem spaţiul posibilelor soluţii. Acesta este format din soluţii, care de fapt înseamnă, aproximativ, 3,6\*109 soluţii. Complexitatea acestui spaţiu al soluţiilor ne obligă să ne îndepărtăm cât mai mult de algoritmii exacţi şi să recurgem la metode care ne oferă soluţii sub-optimale, aşa cum e şi-n cazul algoritmilor genetici.

**Computational results obtained**

Bazându-se pe generarea de numere aleatoare metoda descrisă de mine nu oferă la fiecare rulare aceeaşi soluţie. Din multitudinea de variante obţinute am extras 5, pe care am să la compar cu soluţia oferită de oficiali. Pentru a face comparaţiile am generat aleatoriu 10.000 de locuitori. Probabilitate de generare mai mare o are un locuitor care face parte dintr-un oraş cu populaţia numeroasă, această probabilitate scăzând în acelaşi timp cu scăderea numărului de locuitori.

În tabelul următor avem prezentate 5 rezultate obţinute folosind algorimtul prezentat, dar şi soluţia oferită de autorităţi.

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Oficial solution** | | **1st solution** | | **2nd solution** | | **3th solution** | | **4th solution** | | **5th solution** | |
| **Id** | **City** | **Id** | **City** | **Id** | **City** | **Id** | **City** | **Id** | **City** | **Id** | **City** |
| 1 | Bucuresti | 40 | Roman | 7 | Brasov | 5 | Constanta | 5 | Constanta | 4 | Cluj-Napoca |
| 2 | Iasi | 8 | Galati | 4 | Cluj-Napoca | 1 | Bucuresti | 4 | Cluj-Napoca | 5 | Constanta |
| 3 | Timisoara | 3 | Timisoara | 6 | Craiova | 40 | Roman | 8 | Galati | 7 | Brasov |
| 4 | Cluj-Napoca | 1 | Bucuresti | 3 | Timisoara | 7 | Brasov | 6 | Craiova | 14 | Sibiu |
| 5 | Constanta | 14 | Sibiu | 34 | Deva | 4 | Cluj-Napoca | 14 | Sibiu | 40 | Roman |
| 6 | Craiova | 37 | Sfantu Gheorghe | 40 | Roman | 12 | Arad | 40 | Roman | 3 | Timisoara |
| 7 | Brasov | 36 | Zalau | 8 | Galati | 34 | Deva | 37 | Sfantu Gheorghe | 1 | Bucuresti |
| 8 | Galati | 5 | Constanta | 5 | Constanta | 6 | Craiova | 3 | Timisoara | 8 | Galati |
| 16 | Tg Mures | 6 | Craiova | 1 | Bucuresti | 11 | Braila | 1 | Bucuresti | 6 | Craiova |

Centralizând rezultatele din tabelul 1 obţinem figura 3 cu frecvenţa de apariţie a oraşelor în soluţiile obţinute. Se observă că anumite oraşe apar în toate cele 5 soluţii (Roman, Bucuresti, Constanta, Craiova), în alte cazuri, unde avem câte două oraşe, apropiate geografic (Galati-Braila, Timisoara-Arad), rezultatele indică o diferenţă mare între frecvenţele de apariţie a oraşelor din fiecare caz în parte, iar două dintre oraşele din varianta autorităţilor (Iaşi şi Târgu Mureş) nu apar deloc, poziţia geografică şi vecinătăţile având un cuvânt greu de spus în astfel de cazuri.

O primă problemă pe care am menţionat-o în prima parte a lucrării se referea la faptul că sunt anumite regiuni care nu au nici un spital în varianta găsită de oficiali. În tabelul 2 prezint o centralizare pe regiuni a soluţiilor găsite şi prezentate în tabelul 1.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Oficial solution** | **1st solution** | **2nd solution** | **3th solution** | **4th solution** | **5th solution** |
| RO11 | 1 | 1 | 1 | 1 | 1 | 1 |
| RO12 | 2 | 2 | 1 | 1 | 2 | 2 |
| RO21 | 1 | 1 | 1 | 1 | 1 | 1 |
| RO22 | 2 | 2 | 2 | 2 | 2 | 2 |
| RO31 | 0 | 0 | 0 | 0 | 0 | 0 |
| RO32 | 1 | 1 | 1 | 1 | 1 | 1 |
| RO41 | 1 | 1 | 1 | 1 | 1 | 1 |
| RO42 | 1 | 1 | 2 | 2 | 1 | 1 |

Se observă, că, şi-n variantele găsite folosind algoritmul descris regiunea RO31 rămâne fără spital, explicaţia din primul capitol putând fi luată în considerare în cazul tuturor soluţiilor obţinute. Legat de a doua problemă, cea despre regiunea populată care are un singur spital alocat, prin soluţiile oferite s-a rezolvat o mică parte din problemă. Dacă în varianta iniţială se propunea, în acea regiune, o localitate(cu 362.000 locuitori) aflată la marginea regiunii, aceasta a fost înlăturată, în toate soluţiile găsite, de o localitate ce are în jur de 70.000 de locuitori(Roman). Marele avantaj al localităţii mai mici constă în faptul că se află în centrul regiunii, acaparând o mare parte din populaţia acesteia.

Ultimele două probleme pomenite sunt rezolvate prin apariţia, în soluţiile finale a unor localităţi slab populate (sub 200.000 locuitori), cum ar fi: Roman (5/5), Sibiu (3/5), Sfantu Gheorghe (2/5), Zalau (1/5), Deva (2/5), Arad (1/5)

Validarea acestor soluţii am făcut-o prin generarea aleatoare a 10.000 de locuitori din oraşele din baza de date, probabilitatea ca un cetăţean să fie printre cei 10.000 generaţi depinzând de numărul de locuitori din oraşul din care fac parte. Astfel, cum cât acesta este mai mare, cu atât acest locuitor are mai mari şanse să fie generat. Rezultatele obţinute sunt prezentate în tabelul 2. Şi în acest caz am efectuat 5 rulări, deoarece în spatele testelor se află generatorul de numere aleatoare, ceea ce duce la obţinerea de reultate diferite la fiecare rulare

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Oficial solution** | **1st solution** | **2nd solution** | **3th solution** | **4th solution** | **5th solution** |
| **1st run** | | | | | | |
| **Numărul de km** | 506063 | 487058 | 484869 | 488103 | 484963 | **477718** |
| **Numărul de locuitori de depăşesc 100 km** | 3105 | **2314** | 2550 | 2356 | 2509 | 2561 |
| **Distanţa maximă** | 187 | **163** | 187 | 187 | 187 | 187 |
| **2nd run** | | | | | | |
| **Numărul de km** | 498500 | 490041 | 483053 | 486086 | 483267 | **475077** |
| **Numărul de locuitori de depăşesc 100 km** | 3045 | **2280** | 2559 | 2339 | 2479 | 2538 |
| **Distanţa maximă** | 187 | **163** | 187 | 187 | 187 | 187 |
| **3th run** | | | | | | |
| **Numărul de km** | 504853 | 491003 | 487846 | 494576 | 490150 | **481050** |
| **Numărul de locuitori de depăşesc 100 km** | 3094 | **2230** | 2547 | 2328 | 2511 | 2561 |
| **Distanţa maximă** | 187 | **163** | 187 | 187 | 187 | 187 |
| **4th run** | | | | | | |
| **Numărul de km** | 504602 | 493129 | 483785 | 488131 | 486819 | 479154 |
| **Numărul de locuitori de depăşesc 100 km** | 3112 | **2323** | 2555 | 2348 | 2528 | 2572 |
| **Distanţa maximă** | 187 | **163** | 187 | 187 | 187 | 187 |
| **5th run** | | | | | | |
| **Numărul de km** | 494032 | 478944 | 475905 | 483202 | 476054 | **469194** |
| **Numărul de locuitori de depăşesc 100 km** | 3078 | **2220** | 2504 | 2304 | 2470 | 2529 |
| **Distanţa maximă** | 187 | **163** | 187 | 187 | 187 | 187 |

Din tabelul numărul 3, se observă că soluţia oferită de către autorităţi nu este cea mai bună ţinând cont de indicatorii pe care i-am tratat. Astfel, în afară de ultimul indicator, unde se obţin valori egale cu alte 4 soluţii găsite prin intermediul algoritmul propus (=187 km), în cazul celorlalţi 2 indicatori valorile obţinute la varianta propusă sunt inferioare valorilor obţinute în toate celor 5 variante de soluţii.

**Concluzii**

Algoritmii genetici sunt metode ce se aplică cu succes pentru probleme complexe, pentru a care metodele exacte nu ar putea soluţii. Aceştia se aplică pentru acele probleme combinaţionale care presupun o căutare într-un spaţiu imens al soluţiilor. Şi în cazul problemei noastre, o creştere liniară a datelor de intrare (în cazul nostru, a numărului de oraşe), duce la o creştere exponenţială a numărului de posibile soluţii (cele care se regăsesc în spaţiul soluţiilor). În cazul problemei de faţă am aplicat această metodă şi am obţinut soluţii mai bune decât soluţia recomandată de către autorităţile din România în ceea ce priveşte locaţiile viitoare utilizate pentru construirea unor spitale de dimensiuni mari, la nivel regional. Pentru problema de faţă, ar deveni interesante rezultatele care se obţin în urma aplicării algoritmului pe un set mai mare de oraşe (poate chiar toate care depăşesc 5000 de locuitori). Acceaşi problemă poate fi studiată şi dintr-o perspectivă multicriterială, dar şi prin introducerea unor noi constrângeri, sau chiar modificarea funcţiei obiectiv prin introducerea unor caracteristici noi în calculul acesteia. În cazul de faţă am discutat despre spitale regionale, dar aceeaşi abordare poate fi utilizată în orice altă problemă ce presupune o organizare la nivel de regiuni, cum ar fi: înfiinţarea unor centre logistice, aspecte legate strategii turistice, transporturi, educaţie, etc.

1. Council Of Europe, https://www.coe.int/en/web/congress/regionalisation [↑](#footnote-ref-1)
2. Eurostat, https://ec.europa.eu/eurostat/statistics-explained/index.php?title=Glossary:Region [↑](#footnote-ref-2)
3. Eurostat, https://ec.europa.eu/eurostat/cache/digpub/regions/ [↑](#footnote-ref-3)
4. Rashidian A, Omidvari A, Vali Y, Mortaz S, Yousefi-Nooraie R, Jafari M, et al. The effectiveness of

   regionalization of perinatal care services: a systematic review. Public Health 2014; 128(10):872–85 [↑](#footnote-ref-4)
5. Chih-Yuan Lin & Yue-Chune Lee, Effectiveness of hospital emergency department regionalization and categorization policy on appropriate patient emergency care use: a nationwide observational study in Taiwan, BMC Health Services Research volume 21, Article number: 21 (2021) [↑](#footnote-ref-5)
6. : Ramos MC, Barreto JOM, Shimizu HE,Moraes APGd, Silva ENd (2020) Regionalization for health improvement: A systematic review. PloS ONE 15(12): e0244078. https://doi.org/10.1371/journal.pone.0244078 [↑](#footnote-ref-6)
7. Yousef Chavehpour, Arash Rashidian, Abraha Woldemichael and Amirhossein Takian, Inequality in Geographical distribution ofhospitals and hospital beds in densely populated metropolitan cities of Iran, BMC Health Services Research 19 (2019) [↑](#footnote-ref-7)
8. Mitchell, Melanie (1996). An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press. ISBN 9780585030944. [↑](#footnote-ref-8)
9. Whitley, Darrell (1994). "A genetic algorithm tutorial" . Statistics and Computing. 4 (2): 65–85. CiteSeerX 10.1.1.184.3999.,doi:10.1007/BF00175354 [↑](#footnote-ref-9)
10. Mitchell, Melanie (1996). An Introduction to Genetic Algorithms. Cambridge, MA: MIT Press. ISBN 9780585030944. [↑](#footnote-ref-10)
11. Ionut, Balan, Parallel Hybrid Methods Used In Optimization Problems Solving, The USV Annals of Economics and Public Administration 14 (2 (20)),2014, 168-176 [↑](#footnote-ref-11)
12. Populaţia României pe localităţi, Institutul Naţional de Statistică, 2016, https://insse.ro/cms/sites/default/files/field/publicatii/populatia\_romaniei\_pe\_localitati\_la\_1ianuarie2016\_0.pdf [↑](#footnote-ref-12)
13. https://www.tourismguide.ro/x/distante\_rutiere/ [↑](#footnote-ref-13)
14. https://distanta.ro/ [↑](#footnote-ref-14)