

## Лабораториска вежба бр.3

За оваа лабораториска задача изработив програма (апликација) за локализација на јазли кај безжична сензорска мрежа, со употреба на техниката на 2D трилатерација. Истото го направив преку креирање на 3 класи во програмскиот јазик Јава, од кои параметрите кои се внесуваат за оваа вежба се во класата со име Main а тоа се следните:

N - број на јазли во мрежата

L – должина на областа каде е дистрибуирана мрежата (должина и ширина)

RR - радио опсег

Err – шум на сигналот

f – фракција (процент) на anchor јазли

Покрај оваа класа во која се извршува програмата, во скоп на инплементацијата креирам и две класи едната Node која го означува секој јазел во мрежата и NodeAnchor која ги наследува карактеристиките на класата Node со дополнителен аргумент за тежина кој го искористив при итеративниот пристап за локализација со 3 најрелевантни anchor јазли. Во продолжение се прикажани аргументите на класата Node, која за секој јазел чува и мапа на сосед(клуч) и оддалеченост(вредност) од истиот со шум. При додавање на нов сосед истата се сортира по вредност за секој од двата алгоритми различно, за едниот се сортира опаѓачки по вредноста на оддалеченоста а за другиот опаѓачки за вредноста на тежината на јазелот. На тој начин се поминуваат сите јазли и

- Кај неитеративниот метод само се локализираат оние кои имаат 3 anchor соседи во радио опсегот и за кои се пресметува грешката при локализација
- Кај итеративниот начин едниот метод применува 3 најблиски anchor соседи како неитеративниот начин само што доколку се локализира одреден јазел истиот се додава на листата anchor јазли и се додава во мапата на соседи кај останатите нелокализирани јазли. Додека пак методот со 3 најрелевантни јазли се применува исто така итеративно и се локализираат јазли така што приоритет имаат оние anchor јазли со најмала тежина во мапата на соседи, така доколку еден јазел е локализиран истиот се додава во anchor јазлите со тежина= збир на тежините на 3те јазли со кој е локализиран + 1

```

public class Node {
    private int ID;
    private int x;
    private int y;
    private boolean loc=false; //true ako e lokaliziran
    private double error=0; //greska pri lokalizacija real-predicted distance
    private int xprim=0;
    private int yprim=0;
    private int RR;
    private int Err;
    private TreeMap<NodeAnchor, Double> neighbours;
}

```

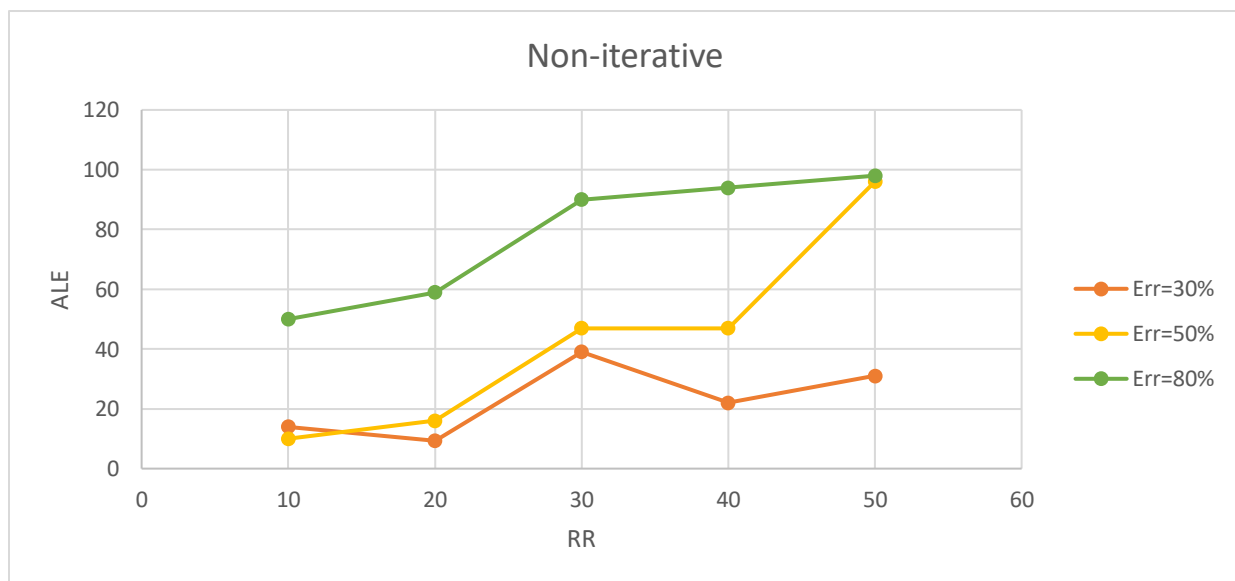
Константни вредности во сите примери се  $N=100$   $L=100$

*Неитеративен алгоритам:*

Точноста од извршената локализација (ALE) во зависност од:

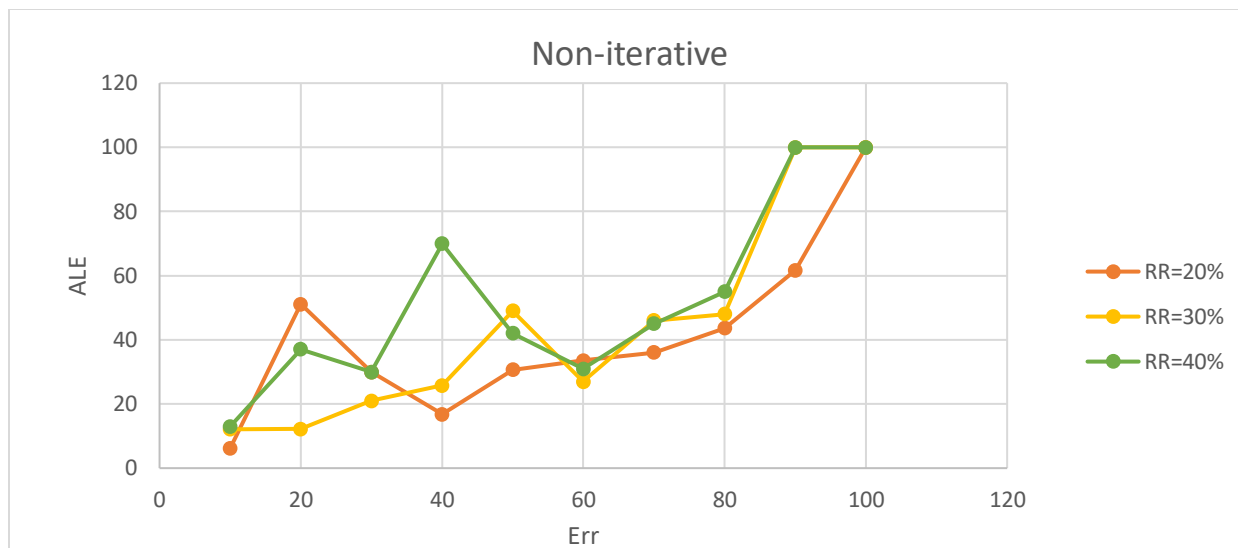
- RR

Преку овој график може да се воочи дека со помал шум се добиваат помали вредности при зголемување на радио опсегот за средната вредност на грешката.



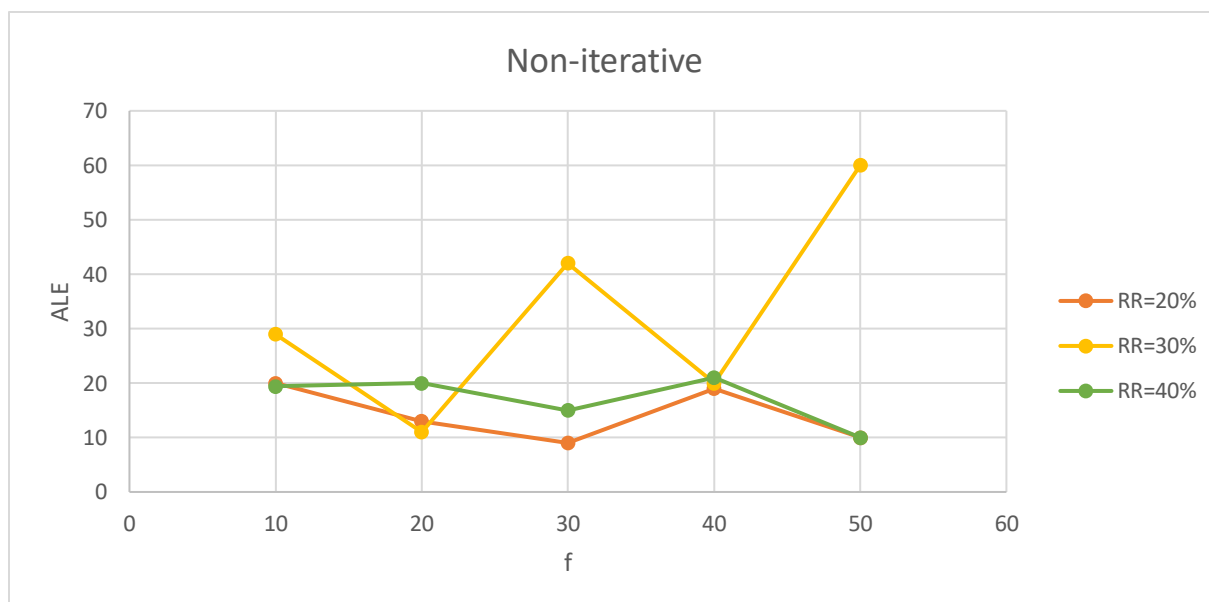
- Err

Преку овој график се воочува дека со зголемување на опсегот и дозволениот процент на шум се зголемува и средната вредност на грешка при локализација.



- f

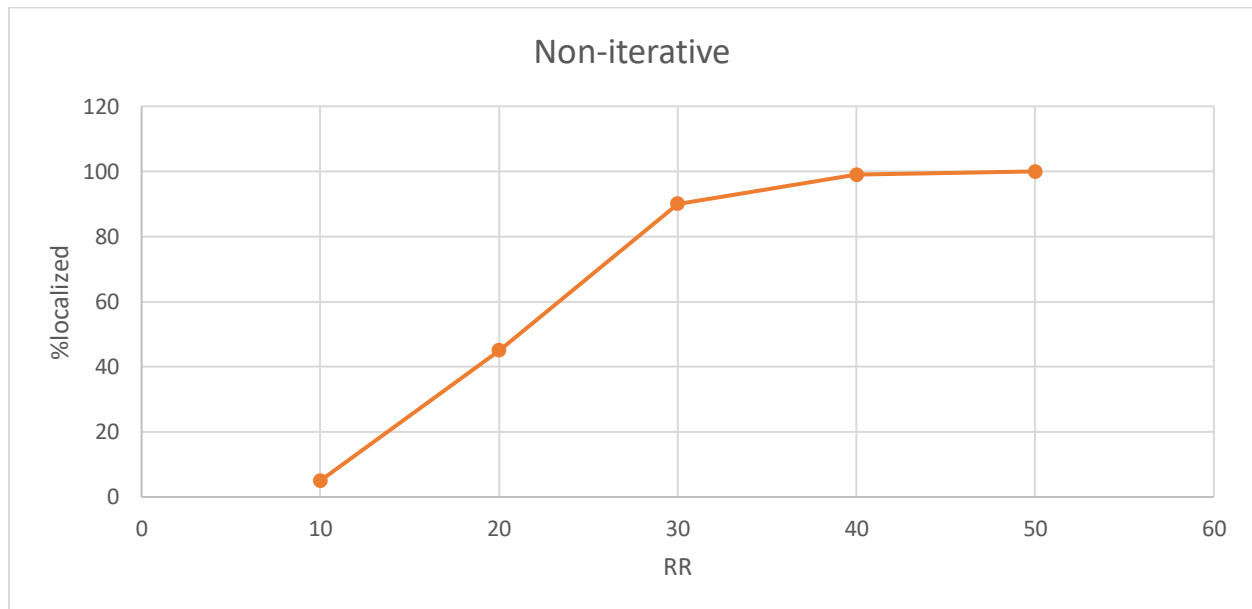
Преку овој график се воочува дека за поголем број на anchor јазли и зголемување на радио опсегот имаме намалување на грешката.



Фракцијата на локализирани јазли во зависност од

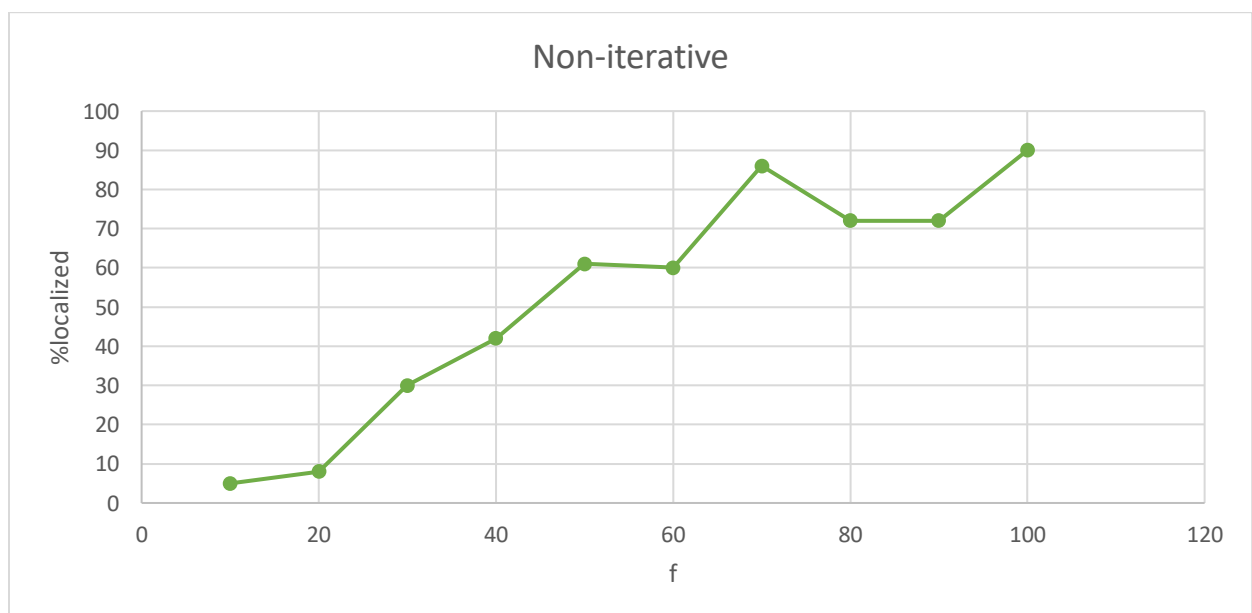
- RR

Прикажана е зависноста на фракција на локализирани јазли изразена во проценти од радио опсегот при користење на неитеративен пристап. Може да се воочи дека како расте радио опсегот на анхор јазлите расте и процентот на локализирани јазли.



- f

Преку овој график може да се воочи дека процентот на локализирани јазли расте со зголемување на бројот на anchor јазли, на таков начин веројатноста да се најде anchor јазел во околината на нелокализиран е поголема.

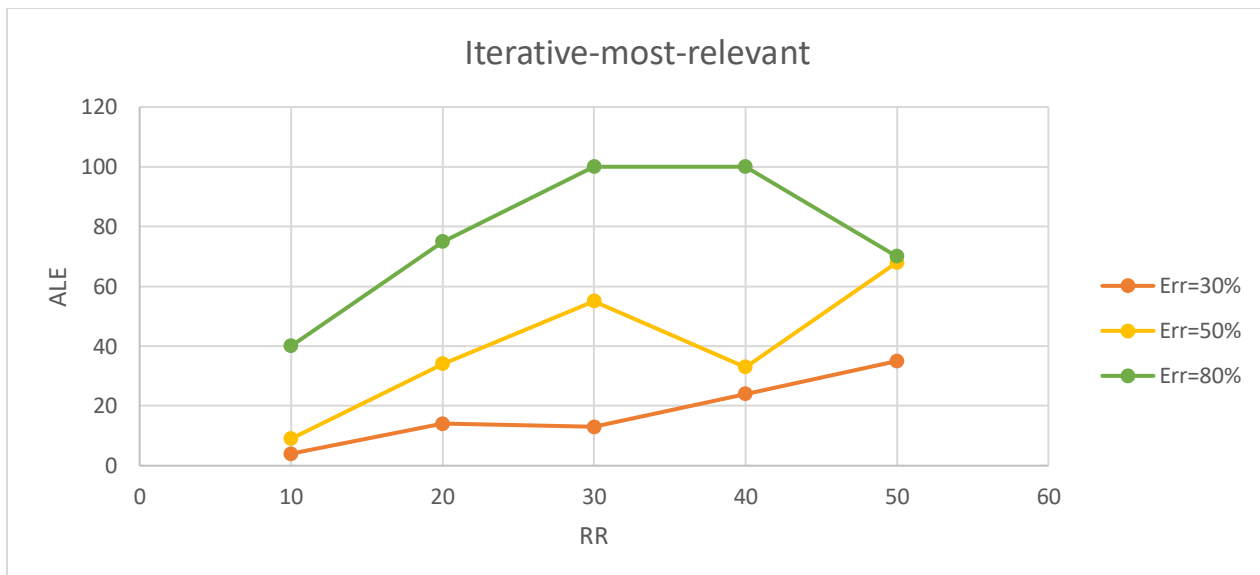
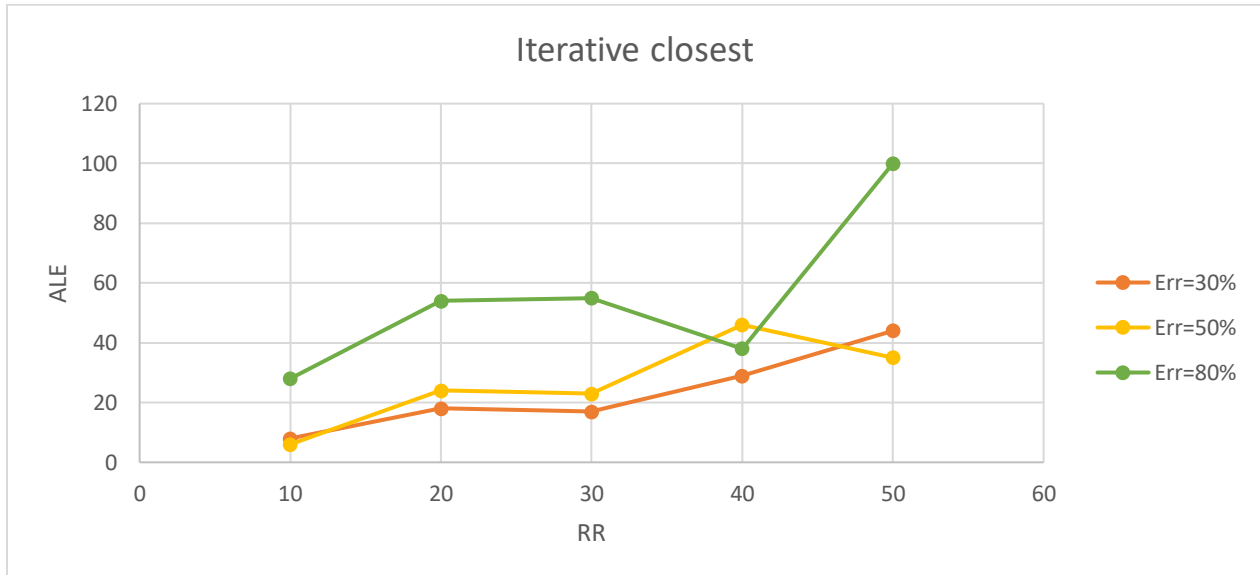


### Итеративен алгоритам:

Точноста од извршената локализација (ALE) во зависност од

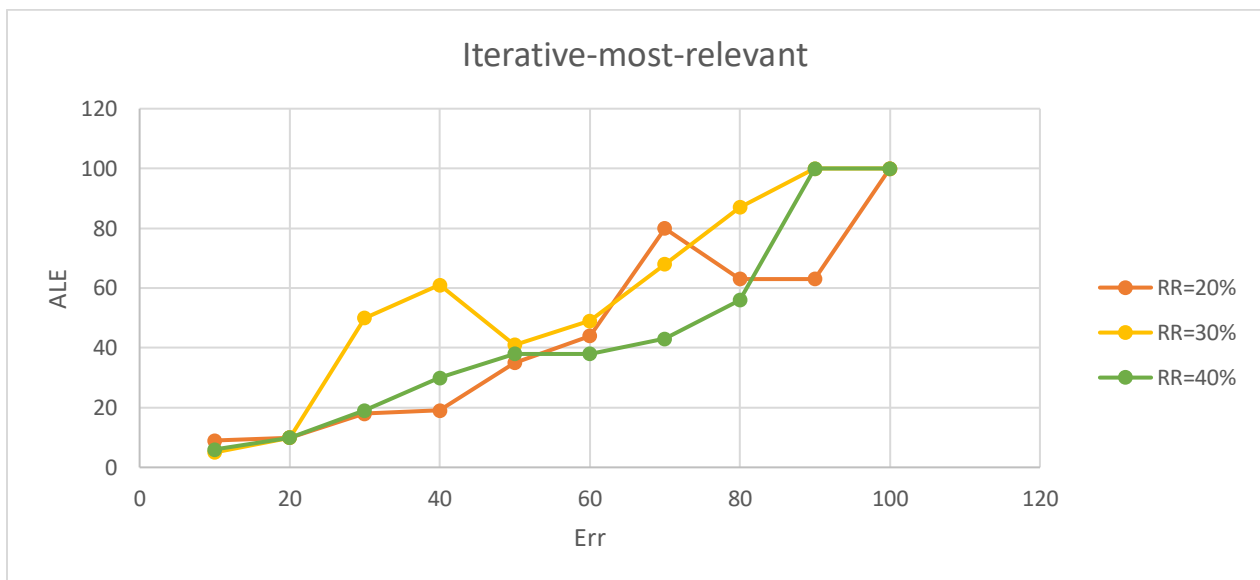
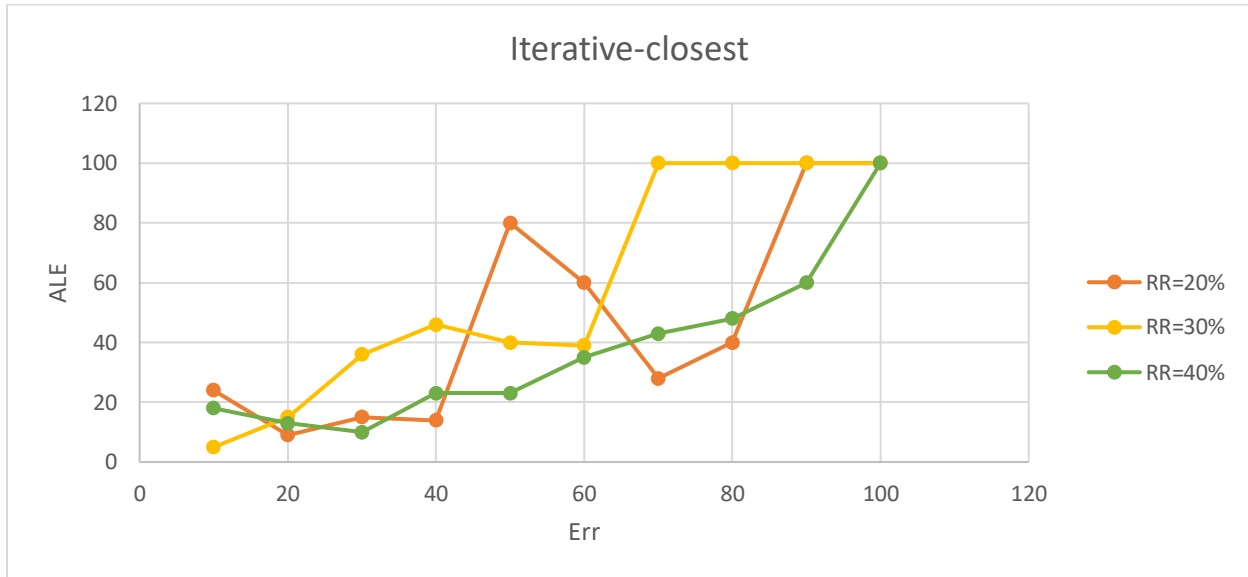
- RR

На следните два графици се воочува дека со зголемување на шумот како и радио опсегот на јазлите се зголемува и средната грешка, но разликите се повоочливи на вториот график во кој се користи евристиката за најрелевантните 3 јазли.



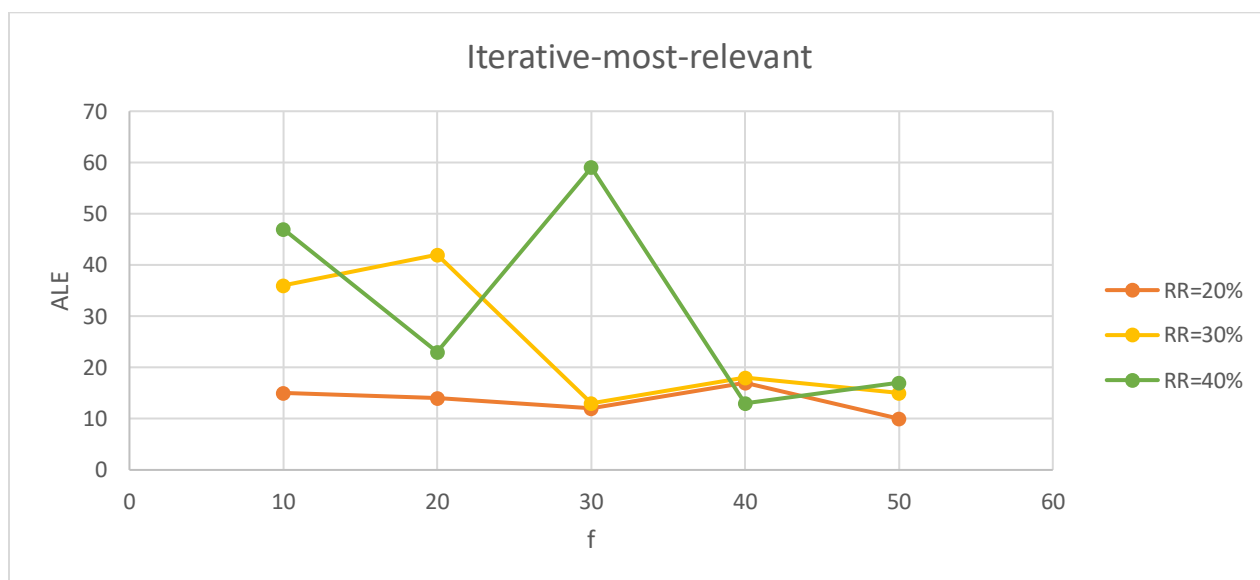
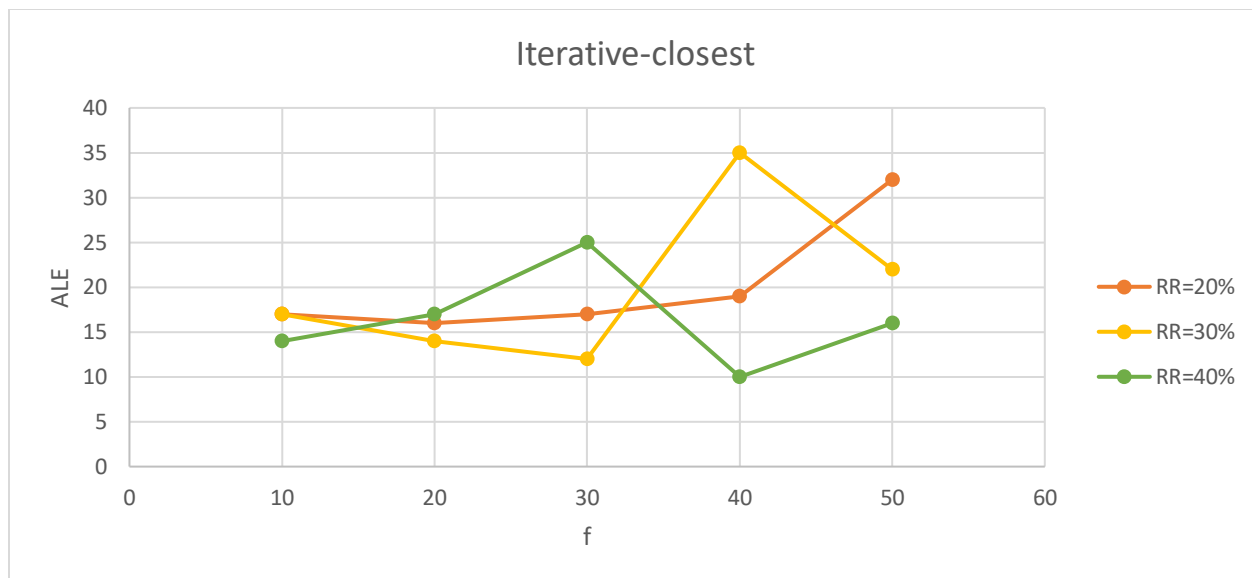
- Err

На следните два графици се прикажани зависностите на дозволената грешка при шум и зголемување на радио опсегот, при што за двете евристики не се забележуваат големи разлики.



- f

На следните два графици се прикажани зависностите при зголемување на anchor јазлите и зголемување на радио опсегот на јазлите и може да се воочи дека за втората евристика со 3 најрелевантни грешката е помала за истите вредности.



Пример излез од програмата:

```
Iterative three most relevant: ALE: 9.999586, Localized nodes(%): 77.5%
Iterative three most relevant: ALE: 10.626215, Localized nodes(%): 65.83333333333333%
Iterative three most relevant: ALE: 291.815, Localized nodes(%): 91.66666666666667%
Iterative three most relevant: ALE: 19.416088, Localized nodes(%): 92.5%
Iterative three most relevant: ALE: 35.452354, Localized nodes(%): 88.33333333333333%
Iterative three most relevant: ALE: 44.216682, Localized nodes(%): 93.33333333333333%
Iterative three most relevant: ALE: 100.17411, Localized nodes(%): 99.16666666666667%
Iterative three most relevant: ALE: 63.22642, Localized nodes(%): 100.0%
Iterative three most relevant: ALE: 63.542027, Localized nodes(%): 100.0%
Iterative three most relevant: ALE: 418.35144, Localized nodes(%): 100.0%
```

Линк за кодот: [https://github.com/GJana/Lab\\_Senzorski\\_Sistemi](https://github.com/GJana/Lab_Senzorski_Sistemi)