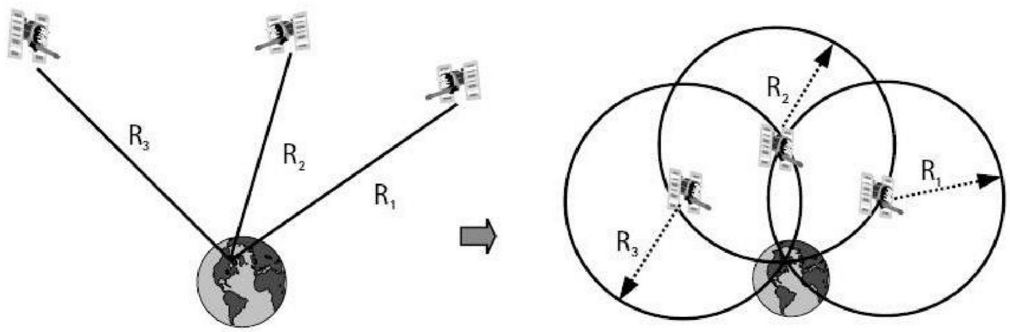
# 2. Gausova eliminacija

## Određivanje koordinata uz pomoć GPS sistema.

Osnovna ideja GPS sistema je varijanta trodimenzionalne triangulacije. Za određivanje položaja na zemljinoj površini su potrebne su samo razdaljine do 3 satelita (ove razdaljine mogu se odrediti vremenom putovanja radio signala). U ovom slučaju, primalac signala se nalazi u preseku 3 sfere, gde svaka sfera ima radijus koji odgovara rastojanju posmatrača i odgovarajućeg satelita (slika 1).




Slika 1. Osnovna ideja kod GPS pozicioniranja

Jednačine koje proizilaze iz opisanih sfera su:

$$(x-x_1)^2+(y-y_1)^2+(z-z_1)^2=r_1{}^2$$
$$(x-x_2)^2+(y-y_2)^2+(z-z_2)^2=r_2{}^2$$
$$(x-x_3)^2+(y-y_3)^2+(z-z_3)^2=r_3{}^2$$
$$(x-x_4)^2+(y-y_4)^2+(z-z_4)^2=r_4{}^2$$

gde $x_i$, $y_i$ i $z_i$ predstavljaju poznate koordinate satelita u svemiru, a $x$, $y$ i $z$ nepoznate koordinate posmatrača na zemlji. Da bi smo došli do sistema linearnih jednačina (da uklonimo kvadrate nepoznatih), oduzimamo prvu jednačinu od ostale tri jednačine. Dobijamo sledeći sistem:

$$2(x_2-x_1)x+2(y_2-y_1)y+2(z_2-z_1)z=r_1^2-x_1^2-r_2^2+x_2^2+y_2^2-y_1^2+z_2^2-z_1^2$$
$$2(x_3-x_1)x+2(y_3-y_1)y+2(z_3-z_1)z=r_1^2-x_1^2-r_3^2+x_3^2+y_3^2-y_1^2+z_3^2-z_1^2$$
$$2(x_4-x_1)x+2(y_4-y_1)y+2(z_4-z_1)z=r_1^2-x_1^2-r_4^2+x_4^2+y_4^2-y_1^2+z_4^2-z_1^2$$

Ovaj sistem se u Python-u može predstaviti matricom *A* i vektorom *b*:

$$A=numpy.array([[2(x_2-x_1)\,2(y_2-y_1)\,2(z_2-z_1)] \quad [2(x_3-x_1)\,2(y_3-y_1)\,2(z_3-z_1)] \quad [2(x_4-x_1)\,2(y_4-y_1)\,2(z_4-z_1)]]) ,$$
$$b=numpy.array([r_1^2-x_1^2-r_2^2+x_2^2+y_2^2-y_1^2+z_2^2-z_1^2] \quad [r_1^2-x_1^2-r_3^2+x_3^2+y_3^2-y_1^2+z_3^2-z_1^2] \quad [r_1^2-x_1^2-r_4^2+x_4^2+y_4^2-y_1^2+z_4^2-z_1^2])$$
$$[x,y,z]=numpy.linalg.solve(A,b)$$

Ovaj primer je moguće rešiti primenom nekog postupka za rešavanje sistema linearnih jednačina, npr. Gausovim postupkom.

**Primer 1: Akumulacija greške kod Gausove eliminacije**

Prilikom modelovanja realnog sveta na računaru uvode se određene aproksimacije: matematički modeli nisu savršena replika realnog sveta, računari na kojima se metode izvršavaju imaju ograničen kapacitet za reprezentaciju brojeva, sami numerički algoritmi daju približna rešenja, itd. Ovaj primer pokazuje kako kod Gausove eliminacije koja sama po sebi daje egzaktno rešenje ipak može da dođe do nezanemarljive greške usled ograničenog kapaciteta za reprezentaciju brojeva. Nakon toga ćemo videti kako malom modifikacijom algoritma (uvođenjem parcijalnog pivotinga) možemo rešiti ovaj problem.

Neka je posmatrač u GPS sistemu izmerio sledeće vrednosti koordinata:

```
s1 = (2088.202, -11757.191, 25391.472)
s2 = (2088.202000000001, -14198.201, 21471.166)
s3 = (35606.985, 94447.027, 9101.379)
s4 = (3966.929, 7362.852, 26388.447)
r = (23204.699, 21585.835, 31364.260, 24966.799)
```

Primetite da $x$ koordinate satelita $s_1$ i $s_2$ imaju veoma bliske vrednosti (razlika između njih je $10^{10}$). Rezultujuća matrica $A$ je:

$$A = 10^5 \begin{bmatrix} 0.000000000000002 & -0.048820200000000 & -0.078406120000000 \\ 0.670375660000000 & 2.124084360000000 & -0.325801860000000 \\ 0.037574540000000 & 0.382400860000000 & 0.019939500000000 \end{bmatrix}$$

U prvom koraku svođenja matrice $A$ na gornju trougaonu se računa $m$:

$$m = A[2,1]/A[1,1] = \frac{0.67037566 \cdot 10^5}{2 \cdot 10^{-10}} = 0.33518783 \cdot 10^{15}$$

Obratiti pažnju da je dobijeno $m$ jako veliko. Nakon množenja prvog reda sa $m$ i njegovog oduzimanja od drugog reda, izračunata matrica $A$ izgleda ovako:

$$A = 10^{18} \begin{bmatrix} 0.000000000000000 & -0.000000000000005 & -0.000000000000008 \\ 0 & 1.643135972212654 & 2.638905949045761 \\ 0.000000000000004 & 0.000000000000038 & 0.000000000000002 \end{bmatrix}$$

Primetite da je došlo do zaokruživanja vrednosti usled predstavljanja jako velikih vrednosti u matrici $A$. Kod $m$ se javila greška se zove *cancellation* (potiranje) – deljenje 2 broja koja su jako blizu jedan drugom – vrednost postaje toliko mala da je besmislena. Množenje prvog reda matrice $A$ sa $m$ i dodavanje na drugi red dovodi do toga da vrednosti u matrici $A$ budu veoma različitih rangova veličina usled čega dolazi do zaokruživanja.

Rešenje koje se dobije primenom Gausove eliminacije i pravo rešenje sitema (respektivno) su:

$$x_{Gauss} = 10^5 \begin{bmatrix} -1.543549314809887 \\ 0.324069117369555 \\ -3.837434547113705 \end{bmatrix}, x_{true} = 10^5 \begin{bmatrix} 2.299178193658841 \\ -0.265409896496146 \\ 0.226286377590409 \end{bmatrix}$$

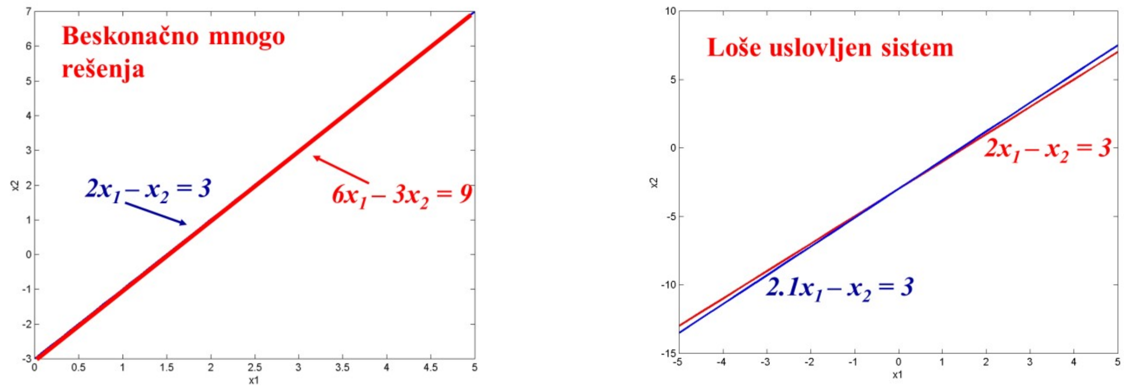
Primetite da je razlika između tačnog rešenja $x_{true}$ i približnog rešenja $x_{Gauss}$ ranga $10^2$. Potrebno je napomenuti da Gausov postupak spada u analitičke postupke rešavanja – sam algoritam dovodi do tačne, a ne približne vrednosti. Uzrok greške moramo tražiti u grešci zaokruživanja koja se akumulirala u toj meri da je dobijeno rešenje veoma pogrešno.

Gubitak tačnosti koji je demonstriran u 1. primeru je moguće smanjiti zamenom redova, tako da se koriste manji množioci $m$ (**Gausova eliminacija sa parcijalnim pivotingom**). Poželjno je da `A[k,k]` bude što veći.

**Primer 2: Loše uslovljen sistem**

Problem je osetljiv ili loše uslovljen ukoliko su relativne promene u izlazima mnogo veće od relativnih promena u ulazima.

Kako utvrditi da je sistem jednačina loše uslovljen?




Slika 2. Loše uslovljen sistem

**Kondicioni broj** funkcije $f(x)$ meri koliko promene ulaza $x$ utiču na promene izlaza $y$. Veliki kondicioni broj znači da male promene ulaza daju velike promene izlaza, odnosno da je funkcija **loše uslovljena**.
Za sistem linearnih jednačina se kondicioni broj računa za matricu $A$ (u Python-u: *numpy.linalg.cond*`(A)`).
Loše uslovljen sistem možemo demonstrirati na slučaju kada su dve jednačine veoma korelirane (druga jednačina je praktično jednaka prvoj pomnoženoj sa nekim koeficijentom). Uzmimo za primer GIS sistem gde se 3 od ukupno 4 uočena satelita nalaze na gotovo istoj poziciji:

```
coef = 1
prag = 0.01

s1 = (2088.202, -11757.191, 25391.472)
s2 = (11092.568, -14198.201, 21471.166)
s3 = (s2[0]*coef + prag, s2[1]*coef + prag, s2[2]*coef + prag)
s4 = (3966.929, 7362.852, 26388.447)

r = (23204.699, 21585.835, 31364.260, 24966.799)
```

Sve ove koordinate su predstavljene uz pomoć 7 značajnih cifara. Izračunati kondicioni broj matrice $A$ je izuzetno velik: $2.28 \cdot 10^6$. Ako rešimo ovaj sistem dobijamo:

$$x_{7\,digits} = 10^3 \begin{bmatrix} 2.089332337320973 \\ -2.846286133242299 \\ -2.463336204003261 \end{bmatrix}.$$

Zaokružimo sada ulazne koordinate na 6 cifara. Rešenje koje sada dobijamo je:

$$x_{6\,digits} = 10^3 \begin{bmatrix} 0.683718836038982 \\ -2.529530021382297 \\ -5.889064158665915 \end{bmatrix}.$$

Dakle, za razliku u ulazima reda veličine $10^{-6}$ dobijamo značajnu razliku u izlazu ranga $10^3$.

**Zadatak 1**

**a)** Primenom funkcije *gauss* rešiti problem određivanja koordinata uz pomoć GPS sistema (motivacioni primer iz materijala) za sledeće izmerene vrednosti koordinata (primetiti da se pri izračunavanju koeficijenata jedn., oduzimanjem odgovarajućih vrednosti koordinata neće dobiti broj blizak $0$):

```
s1 = (2088.202, -11757.191, 25391.472)
s2 = (11092.568, -14198.201, 21471.166)
s3 = (35606.98, 94447.03, 9101.38)
s4 = (3966.929, 7362.852, 26388.447)
r = (23204.699, 21585.835, 31364.260, 24966.799)
```

1. Izračunati matricu $A$ koeficijenata sistema i vektor $b$ slobodnih elemenata
2. Naći rešenje sistema $x$ pozivom funkcije *gauss*, a zatim naći tačno rešenje operatorom \
3. Izračunati apsolutnu grešku

Rezultat:

```
x:   [-154354.93148099   32406.91173696 -383743.45471137]
xt:  [-154354.93148099   32406.91173696 -383743.45471137]
absErr:  [0.     0.     0.]
```

**b)** Ponoviti postupak za sledeće parametre (primetiti da će se pri izračunavanju koeficijenata 1. jedn., oduzimanjem $x_1$ od $x_2$ dobiti broj blizak $0$):

```
s1 = (2088.202, -11757.191, 25391.472)
s2 = (2088.202000000001, -14198.201, 21471.166)
s3 = (35606.98, 94447.03, 9101.38)
s4 = (3966.929, 7362.852, 26388.447)
r = (23204.699, 21585.835, 31364.260, 24966.799)
```

Rezultat:

```
x:   [245760.         -28206.65518901  23665.77777778]
xt:  [229917.81936588 -26540.98964961  22628.63775903]
ae:  [15842.18063412  1665.6655394   1037.14001874]
```

**c)** Ponoviti postupak za parametre pod *b)*, ali upotrebom funkcije *gauss_PP*.

Rezultat:

```
x:   [229917.81936588 -26540.98964961  22628.63775903]
xt:  [229917.81936588 -26540.98964961  22628.63775903]
ae:  [2.91038305e-11 0.00000000e+00 0.00000000e+00]
```

**Zadatak 2**

**a)** Primenom funkcije *gauss* rešiti problem određivanja koordinata uz pomoć GPS sistema za loše uslovljen sistem (primetiti da se koordinate $x_2$ i $x_3$, odnosno $y_2$ i $y_3$, odnosno $z_2$ i $z_3$, vrlo malo razlikuju, odnosno da će rezultovati u dve skoro identične jedn.):

```
coef = 1
prag = 0.01

s1 = (2088.202, -11757.191, 25391.472)
s2 = (11092.568, -14198.201, 21471.166)
s3 = (s2[0]*coef + prag, s2[1]*coef + prag, s2[2]*coef + prag)
s4 = (3966.929, 7362.852, 26388.447)
r = (23204.699, 21585.835, 31364.260, 24966.799)
```

1. Izračunati matricu $A$ koeficijenata sistema i vektor $b$ slobodnih elemenata
2. Izračunati kondicioni broj matrice $A$
3. Vrednosti matrice $A$ i vektora $b$ su date u 7 značajnih cifara. Naći rešenje sistema $x$ pozivom funkcije *gauss*
4. Zaokružiti vrednosti matrice $A$ i vektora $b$ na 6 značajnih cifara (2 decimale), a zatim ponovo naći rešenje sistema $x$ pozivom funkcije *gauss*
5. Izračunati razliku u dobijenim rešenjima

```
kondicioni_broj = np.linalg.cond(A)

xg7 = gauss(A, b)

A6 = np.around(A, 2);
b6 = np.around(b, 2);
xg6 = gauss(A6, b6)

diff = abs(xg7-xg6)
```

Rezultat:

```
kondicioni_broj 2277837.4646321572
xg7:  [-8.06034536e+09  1.81631877e+09 -1.96443817e+10]
xg6:  [-7.90605983e+09  2.38543992e+09 -1.96443814e+10]
diff [1.54285522e+08 5.69121154e+08 2.91804092e+02]
```

**b)** Ponoviti postupak za dobro uslovljen sistem:

```
s1 = (2088.202, -11757.191, 25391.472)
s2 = (11092.568, -14198.201, 21471.166)
s3 = (35606.98, 94447.03, 9101.38)
s4 = (3966.929, 7362.852, 26388.447)
r = (23204.699, 21585.835, 31364.260, 24966.799)
```

Rezultat:

```
cond 67.24859766198261
xg7:  [-154354.91916668   32406.90896205 -383743.4246994 ]
xg6:  [-154354.74772902   32406.91908408 -383743.09574726]
diff [0.17143766 0.01012204 0.32895214]
```