Calculation of GDOP Coefficient

Ing. Martina Mironovova, Ing. Hynek Havliš

Supervisor: Prof. Ing. František Vejražka, CSc., Prof. Ing. Jiří Bíla DrSc.

Abstrakt
Účelem této práce je odvodit koeficient GDOP (Geometrical Dilution of Precision), představující chybu určení polohy uživatele na povrchu Země, a ukázat jeho výpočet na praktické aplikaci. Úkolem je rozmístit až 45 družic po obloze tak, aby se koeficient GDOP minimalizoval a udával tak polohu hledaného objektu na povrchu Země s co nejmenší chybou. Pro optimální rozmístění družic na obloze je použito genetického algoritmu.

Abstract
The purpose of this work is to derive a GDOP coefficient (Geometrical Dilution of Precision), that determines the error of the position of observer on the Earth surface, and to show a calculation of the coefficient in practical application. The task is to distribute up to 45 satellites above the sky so that the GDOP coefficient is minimized and determines position of the searched object on Earth surface with minimal error. Genetic algorithm is used to determine optimal distribution of all satellites above the sky.

Keywords: GDOP, satellite navigation, genetic algorithm

1. Introduction

The main motivation for this work is a derivation of GDOP coefficient, practical example of its calculation and use of GDOP coefficient for determining positions of arbitrary number of satellites above the sky. MATLAB software is used for the calculation and simulation of satellites positions and for GDOP coefficient calculation. The whole task is divided into three sections. The first part is theoretical and shows the derivation of GDOP coefficient. In the second part the use of a genetic

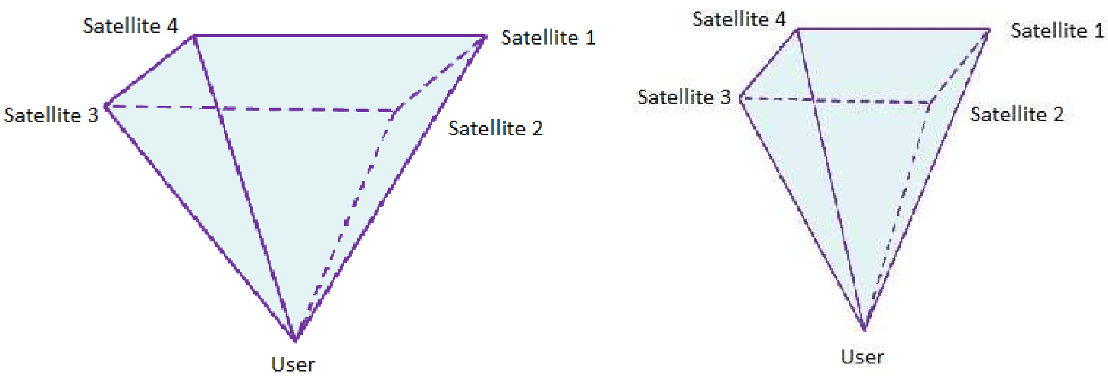


Figure 1: Imaginary pyramids with four satellites in corners of the base and user placed at the tip. Pyramid on the left shows satellite distribution that yields a lower value of GDOP coefficient and is better than configuration of satellites distributed according to the pyramid on the right that shows worse (higher) value of the coefficient.

algorithm is shown. The algorithm is used to find the most optimal setup of satellite positions so that the lowest value of GDOP is reached. The third part includes calculation of the coefficient due to positions of satellites previously calculated by a genetic algorithm.

A GDOP coefficient (Geometric Dilution of Precision) is used in satellite navigation and positioning and represents a ratio of the position error to the range error [1]. The coefficient reflects the dilution of precision in position in three dimensions (PDOP) and dilution of precision in time (TDOP). To compute these four dimensions – position in *x*,*y*,*z* and time – four satellites are needed. The receiver position is computed from satellites positions, the measured pseudo-ranges and receiver position estimate.

Let's imagine that a square pyramid is formed by lines joining four satellites with the receiver placed at the tip of the pyramid (see *Figure 1*). The volume of the shape described by the unit-vectors from the receiver to the satellites used in a position fix is inversely proportional to GDOP. The larger is the volume of the pyramid, the better (lower) the value of GDOP coefficient is. Reversely, the smaller volume of the pyramid is, the worse (higher) the value of GDOP will be. Similarly, the greater number of satellites is used for position estimation, the better the value of GDOP coefficient is [1], [2].

The importance of satellite distribution can be also seen in *Figure 2* that shows three cases how two satellites and their pseudo-ranges determine the area of possible occurrence of the receiver (user). Each satellite has a pseudo-range represented by sector of a circle. Real position of the user is in the intersection of real satellite ranges (circle with a center in each satellite). However, in reality these ranges shape not just

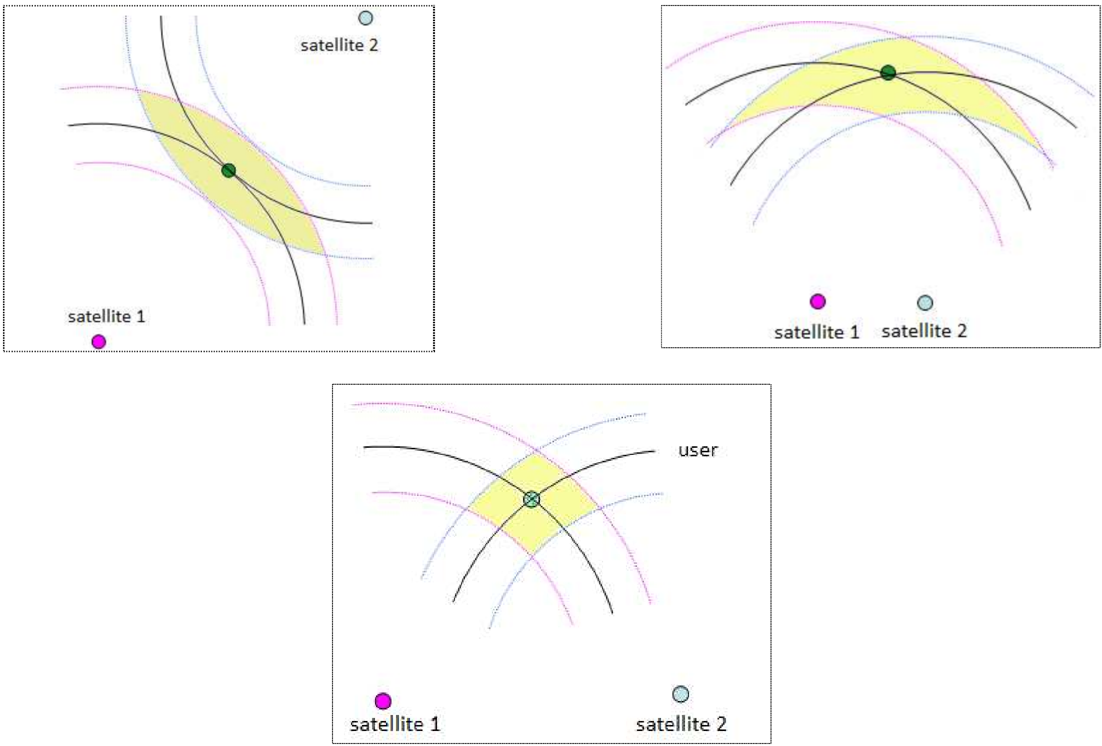


Figure 2: Distribution of two satellites above the receiver (user) - top left and right image show bad distribution with large pseudo-ranges, bottom image displays optimal pseudo-range.

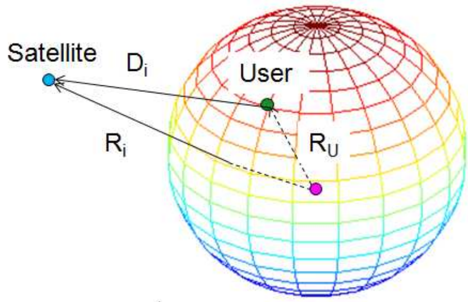


Figure 3: Geometry of vectors for position determination

one point in the place where they cross, but an area instead. This pseudo-range can vary according to the position of satellites above the sky. While the angle between satellites is too wide, the pseudo-range where user can be present is large rather long. The similar situation occurs while the angle between satellites is too small. Optimally a smallest possible square area is required.

2. Derivation of GDOP Coefficient

For the position of the user the following equation applies (see *Figure 3*) [3]:

$$\mathbf{R_U} = \mathbf{R_i} - \mathbf{D_i} \tag{2.1}$$

$\mathbf{R_U}$ – position vector of the user (unknown) center of Earth - user
$\mathbf{R_i}$ – vector from the Earth's center to the satellite
$\mathbf{D_i}$ – vector from the user to the satellite

The equation (2.1) is multiplied by vector $\mathbf{e_i}$, that is a unit vector in direction of $\mathbf{R_i}$:

$$\mathbf{e_i R_U} = \mathbf{e_i R_i} - \mathbf{e_i D_i} \tag{2.2}$$

For the vector, its absolute value and a unit vector in its direction the following proposition is valid:

$$\mathbf{eR} = |\mathbf{R}| \tag{2.3}$$

Now for the equation (2.2) equation (2.3) is applied, and this operation gains following:

$$\mathbf{e_i R_U} = \mathbf{e_i R_i} - |\mathbf{D_i}| \tag{2.4}$$

Vector $\mathbf{D_i}$ magnitude (its absolute value) is a distance user - satellite. The equation (2.4) conforms (2.3) equation between numbers, not vectors.

For the distance $|\mathbf{D_i}|$ the definition applies:

$$|D_i| = \rho_i\text{-}B_u\text{-}B_i \tag{2.5}$$

In this equation:

ρ_i - is so called pseudo-range (pseudo-distance user – satellite that corresponds to the duration of signal propagation from the satellite to the user). Term pseudo-range means that this distance is not determined by the duration of signal propagation from the satellite to the user. It is due to the non-constant properties of the atmosphere (permittivity, permeability) and/or due to the reason of non-propagation of the signal to the user (bounces from the buildings, mountains, etc.).

B_u – is a user's time offset from the theoretically correct time
B_i – is a satellite time offset from the theoretically correct time
The equation (2.5) is established into (2.4) and the following equation is obtained:

$$\begin{aligned} \mathbf{e_i R_U} &= \mathbf{e_i R_i} - \rho_i + B_u + B_i \\ \mathbf{e_i R_U} - B_u &= \mathbf{e_i R_i} - \rho_i + B_i \end{aligned} \tag{2.6}$$

The equation (2.6) applies for one user and theoretically for any satellite (arbitrary number of satellites).

Let's mark coordinates of the directional vector from the user to the i^th^ satellite as e_{i1}, e_{i2}, e_{i3}.

The equation (2.6) can be transcribed for more satellites. It will be shown that for the explicit determination of position it is necessary to have three satellites. If more satellites are available, the optimal solution will be found in accordance by a least square method. If the user time offset B_U is necessary to determinate, at least four satellites will be needed.

The equation (2.6) can be transcribed for more satellites:

$$\begin{aligned} e_{11}\mathrm{R}_{U1} + e_{12}\mathrm{R}_{U2} + e_{13}\mathrm{R}_{U3} - B_U &= e_{11}R_{11} + e_{12}R_{12} + e_{13}R_{13} + B_1 - \rho_1 \\ e_{21}\mathrm{R}_{U1} + e_{22}\mathrm{R}_{U2} + e_{23}\mathrm{R}_{U3} - B_U &= e_{11}R_{11} + e_{12}R_{12} + e_{13}R_{13} + B_1 - \rho_2 \\ e_{31}\mathrm{R}_{U1} + e_{32}\mathrm{R}_{U2} + e_{33}\mathrm{R}_{U3} - B_U &= e_{11}R_{11} + e_{12}R_{12} + e_{13}R_{13} + B_1 - \rho_3 \\ e_{41}\mathrm{R}_{U1} + e_{42}\mathrm{R}_{U2} + e_{43}\mathrm{R}_{U3} - B_U &= e_{11}R_{11} + e_{12}R_{12} + e_{13}R_{13} + B_1 - \rho_4 \end{aligned} \tag{2.7}$$

If R_{Ui} is determined then the position of the user is known and the task is being solved.

The system (2.7) is obviously possible to rewrite and solve out using a matrix form. It can be marked that:

$$G_U = \begin{bmatrix} e_{11} & e_{12} & e_{13} & 1 \\ e_{21} & e_{22} & e_{23} & 1 \\ e_{31} & e_{32} & e_{33} & 1 \\ e_{41} & e_{42} & e_{43} & 1 \\ & \dots & & \end{bmatrix} \qquad X_U = \begin{bmatrix} R_{U1} \\ R_{U2} \\ R_{U3} \\ -B_U \end{bmatrix}$$

$$A_U = \begin{bmatrix} e_{11} & e_{12} & e_{13} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & e_{21} & e_{22} & e_{23} & 1 & 0 \\ & & & & & & & & \dots \\ & & & \dots\dots & & & & & \\ & & & & & & & & \end{bmatrix}$$

$$S = [R_{11} \quad R_{12} \quad R_{13} \quad B_1 \quad R_{21} \quad R_{22} \quad R_{23} \quad B_2 \quad \dots]^T$$

$$\rho = [\rho_1 \quad \rho_2 \quad \rho_3 \quad \rho_4 \quad \dots]^T$$

Matrix dimensions are following (n = number of satellites, n ≥ 4).

G_u <n × 4>
X_u <4 × 1>

A_u <n × 4n>
S <4n × 1>
ρ <n × 1>

Using these assumptions the equation (2.7) can be re-written in a matrix form as follows:

$$\mathbf{G_U X_U} = \mathbf{A_u S} - \boldsymbol{\rho} \tag{2.8}$$

Searched position of the user and the time offset of his receiver are included in vector X_u, and the resolution of the system (2.8) for X_u is our task.

If we had exactly four satellites the solution of (2.8) would be following:

$$\mathbf{X_U} = \mathbf{G}_U^{-1}(\mathbf{A_u S} - \boldsymbol{\rho}) \tag{2.9}$$

Providing that the inverse of G_u matrix exists – if all satellites are distributed “correctly”. It would not be calculated while all satellites are in one position or in one line consecutively. Both situations are universally irrational. Positions of satellites obviously influence the accuracy of user position determination, because pseudo-ranges are burdened by random errors.

If the signal received comes from more than four satellites, the equation (2.8) can be converted by adding vector of unknown errors **e** so that equations are algebraically correct:

$$\mathbf{G_U X_U} + \mathbf{e} = \mathbf{A_u S} - \boldsymbol{\rho} \tag{2.10}$$

For the error vector **e** applies the following:

$$\mathbf{e} = \mathbf{A_u S} - \boldsymbol{\rho} - \mathbf{G_U X_U} \tag{2.11}$$

The system (2.11) can be calculated using a least squares method – minimizing the sum of quadrate of errors, i.e. sum of quadrates of vector **e** elements.

The sum of quadrates of errors is given by a scalar product:

$$P = e^T e = (\mathbf{A_u S} - \boldsymbol{\rho} - \mathbf{G_U X_U})^T (\mathbf{A_u S} - \boldsymbol{\rho} - \mathbf{G_U X_U}) \tag{2.12}$$

The equation (2.12) can be modified using an operator for transpose and by multiplication:

$$\begin{aligned} \mathbf{P} &= ((\mathbf{A_u S} - \boldsymbol{\rho})^\mathbf{T} - \mathbf{X_U^T G_U^T})(\mathbf{A_u S} - \boldsymbol{\rho} - \mathbf{G_U X_U}) \\ \mathbf{P} &= (\mathbf{A_u S} - \boldsymbol{\rho})^\mathbf{T}(\mathbf{A_u S} - \boldsymbol{\rho}) - (\mathbf{A_u S} - \boldsymbol{\rho})^\mathbf{T}\mathbf{G_U X_U} - \mathbf{X_U^T G_U^T}(\mathbf{A_u S} - \boldsymbol{\rho}) + \mathbf{X_U^T G_U^T G_U X_U} \end{aligned} \tag{2.13}$$

While applying a transpose function the following equation was used: $\mathbf{(AB)^T} = \mathbf{B^T A^T}$

Due to the previously given dimensions of vectors and matrices in equation (2.13) it is obvious that members $(\mathbf{A_u S} - \boldsymbol{\rho})^\mathbf{T}\mathbf{G_U X_U}$ and $\mathbf{X_U^T G_U^T}(\mathbf{A_u S} - \boldsymbol{\rho})$ are scalars (numbers) and are equal. Using this assumption the equation (2.13) can be re-written as follows:

$$\mathbf{P} = (\mathbf{A_u S} - \boldsymbol{\rho})^\mathbf{T}(\mathbf{A_u S} - \boldsymbol{\rho}) - 2\mathbf{X_U^T G_U^T}(\mathbf{A_u S} - \boldsymbol{\rho}) + \mathbf{X_U^T G_U^T G_U X_U} \tag{2.14}$$

In this moment we are looking for such X_U so that the sum of quadrates of errors P is minimal. We can derive the equation (2.14) by X_U and set it equal to zero in order to find optimal X_U:

$$\frac{\partial \mathbf{P}}{\partial \mathbf{X_U}} = -\,\mathbf{2G_U^T(A_uS-\rho) + 2G_U^TG_UX_U} \tag{2.15}$$

In derivation we have used following matrix equations:

$$\frac{\partial}{\partial \mathbf{x}}\left(\mathbf{x^TAy}\right) = \mathbf{Ay}, \frac{\partial}{\partial \mathbf{x}}\left(\mathbf{x^TAx}\right) = \mathbf{Ax + A^Tx}$$

Now we set the derivation equal to zero and calculate X_U.

$$\begin{aligned} 0 &= -\,\mathbf{2G_U^T(A_uS-\rho) + 2G_U^TG_UX_U} \\ \mathbf{G_U^T(A_uS-\rho)} &= \mathbf{G_U^TG_UX_U} \\ \mathbf{X_U} &= (\mathbf{G_U^TG_U})^{-1}\mathbf{G_U^T(A_uS-\rho)} \end{aligned} \tag{2.16}$$

Arrangements are proceeded on condition that inverse of matrix $\mathbf{G_U^TG_U}$ exists.

From now on we will consider about the error in determination of position GDOP from satellites using the equation (2.16).

Covariance matrix of position errors X is defined as follows:

$$\operatorname{cov}\delta X_U = E((X_U - EX_U)(X_U - EX_U)^T) \tag{2.17}$$

where E is the operator of the mean value. We assume that the mean value of the position error is equal to zero. With this assumption the equation (2.17) can be simplified:

$$\operatorname{cov}\delta X_U = E(X_U X_U^T) \tag{2.18}$$

while knowing that equation for X_u is known (2.16). The equation (2.16) can be substituted into (2.18):

$$\operatorname{cov}\delta X_U = E((\mathbf{G_U^TG_U})^{-1}\mathbf{G_U^T(A_uS-\rho)(A_uS-\rho)}^T\,\mathbf{G_U}\ (\mathbf{G_U^TG_U})^{-1T}) \tag{2.19}$$

Matrices $\mathbf{G_U}$ consist of measured values and that is why we can impute it before the operator of mean value:

$$\operatorname{cov}\delta X_U = (\mathbf{G_U^TG_U})^{-1}\mathbf{G_U^TG_U}\ (\mathbf{G_U^TG_U})^{-1T}\,E\mathbf{(A_uS-\rho)(A_uS-\rho)}^T \tag{2.20}$$

Because we are interested in calculating the position error from the geometry distribution of satellites (<u>G</u>DOP), we can define the following:

$$E\mathbf{(A_uS-\rho)(A_uS-\rho)}^T = \mathrm{I} \tag{2.21}$$

In the equation above, I represents the unit matrix 4×4. Under this assumption for covariance matrix of errors, the following equation is valid:

$$\operatorname{cov}\delta X_U = (\mathbf{G_U^TG_U})^{-1}\mathbf{G_U^TG_U}\ (\mathbf{G_U^TG_U})^{-1T} = (\mathbf{G_U^TG_U})^{-1T} \tag{2.22}$$

Because the first two matrices are mutually inverse:

$$(\mathbf{G_U^TG_U})^{-1}\mathbf{G_U^TG_U} = I$$

Because the covariance matrix is symmetrical, the rest of the equation (2.22) can be simplified as follows:

$$\operatorname{cov}\delta X_U = (\mathbf{G_U^TG_U})^{-1T} = (\mathbf{G_U^TG_U})^{-1} \tag{2.23}$$

This covariance appears as:

$$\left(G_u^T G_u\right)^{-1} = \begin{array}{c} \\ X \\ Y \\ Z \\ Time \end{array} \begin{array}{c} \begin{array}{cccc} X & Y & Z & Time \end{array} \\ \begin{bmatrix} \sigma_{xx}^2 & \sigma_{xy}^2 & \sigma_{xz}^2 & \sigma_{xt}^2 \\ \sigma_{yx}^2 & \sigma_{yy}^2 & \sigma_{yz}^2 & \sigma_{yt}^2 \\ \sigma_{zx}^2 & \sigma_{zy}^2 & \sigma_{zz}^2 & \sigma_{zt}^2 \\ \sigma_{tx}^2 & \sigma_{ty}^2 & \sigma_{tz}^2 & \sigma_{tt}^2 \end{bmatrix} \end{array} \tag{2.24}$$

The diagonal values in matrix (2.24) represent the variance of the estimated user position in each axis and in the user time offset. The individual factors of GDOP are given as follows:

$$HDOP = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2} \tag{2.25}$$

$$VDOP = \sigma_{zz} \tag{2.26}$$

$$PDOP = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2} \tag{2.27}$$

$$TDOP = \sigma_{tt} \tag{2.28}$$

$$GDOP = \sqrt{\sigma_{xx}^2 + \sigma_{yy}^2 + \sigma_{zz}^2 + \sigma_{tt}^2} \tag{2.29}$$

Estimates of errors in user position or in user time are given as a product of GDOP factors and estimates of errors in range measurements.

3. Finding Positions of Satellites

For the calculation of GDOP coefficient itself it is necessary to know positions of satellites on orbit around the Earth. Because from the theory and derivation from the Chapter 2 it appears that the minimal value of the coefficient GDOP is while all satellites are distributed equally above the sky – their mutual distances are equal [3].

The sky can be approximated as a hemisphere with a center placed in the center of Earth. The estimated position of the object is placed on the Earth’s surface, however, satellites are placed on geostationary orbit with a radius approximately 7-times greater than radius of Earth. Thus the above listed assumption can be applied.

Since the satellite cannot be placed exactly above the horizon, because the emitted waves from the emitter may not reach the receiver due to the surface asperity, it is necessary to set up the minimal elevation angle. Under this angle the satellite cannot be placed. Practically this elevation angle equals to 5º. A schematic sketch displayed in *Figure 4* shows the estimation of total elevation angle. Due to the approximation described above it is necessary to add a correction of 8.7º.

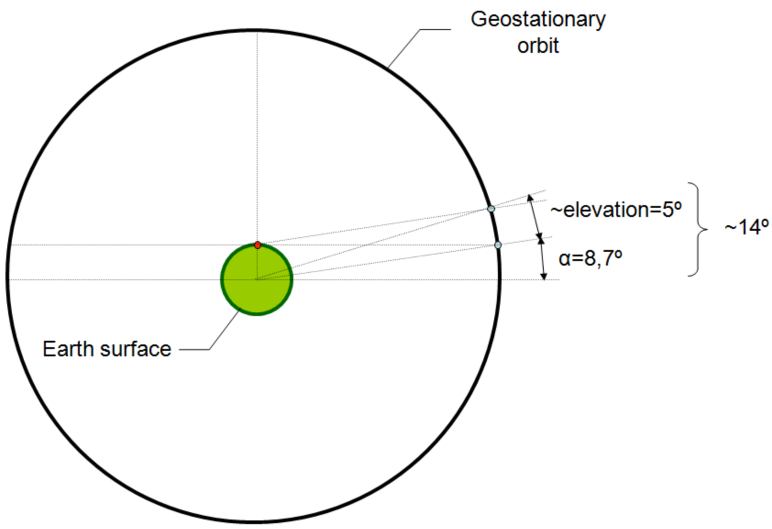


Figure 4: Schematic sketch displaying the estimation of total elevation angle equal to 14° (degrees) approximately.

For equal distribution of satellites above the sky multiple heuristics can be used – from the planary coverage of the area, through random selection up to algorithms of artificial intelligence [4]. In this work a method of genetic algorithms was selected. This method is inspired by Darwin's theory of descent.

The algorithm uses a group of solutions between which it selects the best ones. These selected solutions are combined between each other (so called crossover) and this leads to newly incurred solutions. In the meantime the week solutions (worst ones) are being eliminated. In order to prevent the group of solutions to be restrained just to combination of already existing solutions, with known probability the mutation of selected specimens occurs in every step of the algorithm.

The effort is to enhance the population in every step of the algorithm that can lead to the retrieval of optimal solution, or solution close to the optimal one.

For the estimation of the quality of the solution so called fitness function can be used. This function can take arbitrary form. In this example the fitness function is a function of distances between two closest satellites and is specified by following formula:

$$d = \min_{i,j\; i\neq j}\left(\arccos\left(x_i x_j + y_i y_j + z_i z_j\right)\right)$$

The aim is to maximize this function.

Positions of satellites of concrete solution are for the purpose of algorithm coded into so called genome, or chromosome. Such a chromosome contains of genes and each gene represents one satellite. Values of the gene represent the spherical coordinates of the satellite. At the beginning of the algorithm these spherical coordinates are selected randomly for each specimen.

The population matrix goes through four phases in the algorithm. Firstly the elitism is applied where *k* specimens with the best value of fitness function are selected from the population. These specimens are isolated from the population and preserved to ensure the conservation of the present best solution. In the second phase of the algorithm the crossover of two selected genomes occurs. The place of the crossover process is selected randomly and a new specimen originates. This new specimen picks

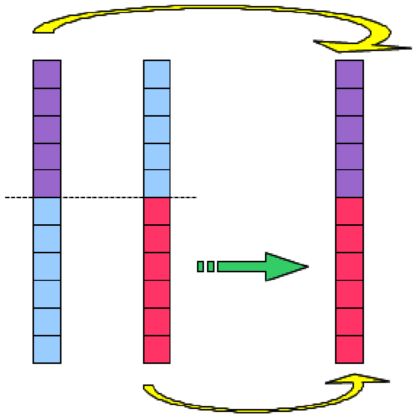


Figure 5: Process of crossover of two specimens and origination of a new specimen

up part of the information from the first specimen and part of information from the second specimen. This process is shown in *Figure 5*.

In the mutation phase the random change of one gene in the genome occurs. During the elimination phase the worst specimen is suspended from the population. After this phase the specimens that were at the beginning preserved (elitism phase) are returned back into the population. All four phases repeat again, as is shown in the following pseudo code:

```
Population = Generate_random_opulation(population_size)
while end condition not satisfied do
      // Preserve elite
      Elite = Pop_elite_from_population(elite_size)

      // Reproduction
      Indiv_1 = Get_random_individual_from_population()
      Indiv_2 = Get_random_individual_from_population()
      New_indiv = Apply_crossover(Indiv_1, Indiv_2)
      Add_indiv_to_population(New_indiv)

      // Mutation
      Mutatate = Random_number
      if Mutate <= Mutation_probability
            Mutated = Get_random_individual_from_population()
            Mutated = Apply_mutation(Mutated)
      end

      // Elimination
      Eliminate_worst(elimination_count)

      // Restore elite
      Return_elite_to_population

      // Remember best solution
      Currently_best = Get_best_individual()
      if Evaluate(Currently_best) > Evlauate(Best)
            best = Currently_best
      end
end
return Best;
```

Algorithm is terminated when all *k* iterations run over. In order to obtain good results it is necessary to select sufficient amount of iterations or select the termination condition so that the value of fitness function will not change more than is the defined certain value.

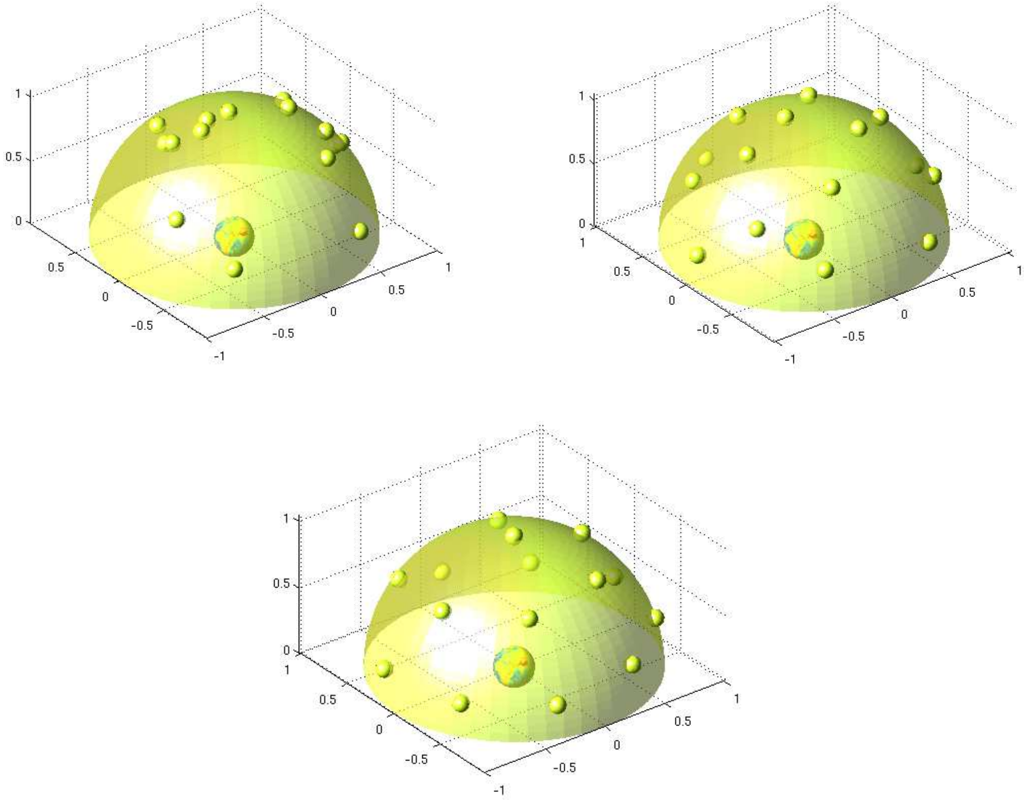


Figure 6: Positions of satellites during their distribution (top left: 1 iteration, top right: 500 iterations, bottom: 10,000 iterations)

For the calculation of the algorithm described above and for the visualization of the results the MATLAB application can be used. *Figure 6* displays three plots – in each there is an Earth displayed with the imaginary hemisphere with radius of geostationary orbit. On this imaginary hemisphere individual satellites are displayed. The figure shows the process of the algorithm when the solution reaches its optimum. Firstly the system starts from the random configuration of satellites that are being slowly distributed in the way that the minimum distance function is maximized. After 10,000 iterations satellites are distributed equally above the hemisphere.

Number of satellites and elevation angle can be changed arbitrarily. Also the set up of the algorithm can be changed (number of iterations, number of reproduced specimens, size of the population, mutation probability).

4. Calculation of GDOP Coefficient

Calculation of GDOP coefficient was described in detail in Chapter 2. Genetic algorithm described in Chapter 3 allows obtaining positions of satellites above the sky. For GDOP coefficient calculation MATLAB and data obtained from previous calculations by genetic algorithm can be used. The output of this part of computer program is a GDOP coefficient and its parameters: PDOP (dilution of precision in position in three dimensions), HDOP (dilution of precision in two horizontal dimensions) and TDOP (dilution of precision in time).

MATLAB program environment allows user to optimize its own graphical user interface (GUI) for work facilitation with a computer program. In the top left corner of the application window of GUI (please see *Figure 7*) it is possible to enter three variables – input parameters i.e. number of satellites above the sky, elevation angle over the horizon and number of iterations of genetic algorithm. A pushbutton labeled ***Start výpočtu*** runs the computer program which according to entered input parameters calculates the optimal distribution of satellites and determines values of coefficient. These values are displayed in corresponding text fields in GUI window.

As an additional output two plots are displayed on the right hand side of the GUI window. Top plot shows a polar chart of satellites above the sky, the bottom one displays the same satellites distributed in 3-D view.

Figure 7 displays the application window and results for distribution of 45 satellites including DOP values. From this example it can be observed that satellites are distributed evenly above the hemisphere, however to reach such a performance at least 160,000 iterations are necessary to reach a good distribution at the expense of calculation time.

5. Results

Examples in previous chapters show that program using a genetic algorithm is able to distribute variable amount of satellites evenly on the hemisphere and that the distribution quality is dependent on number of iterations used. The higher the number of satellites, the higher the value of iterations is needed in order to obtain a good solution.

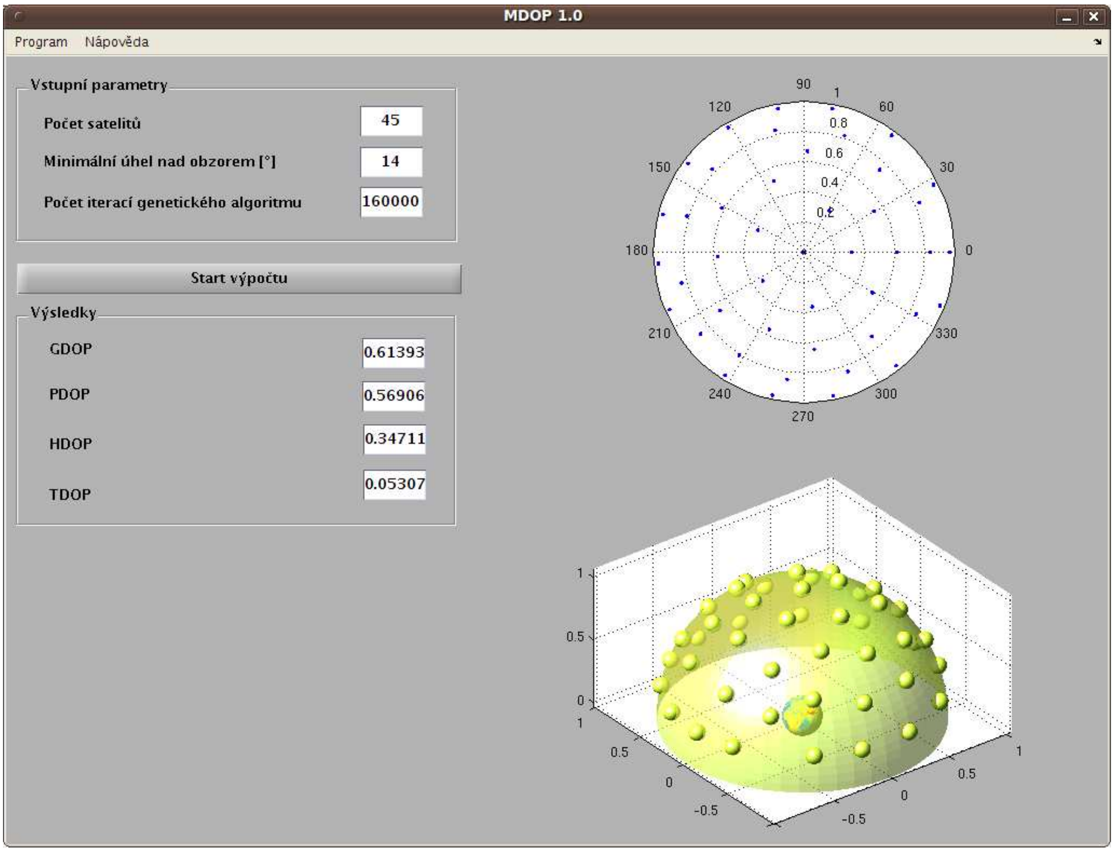
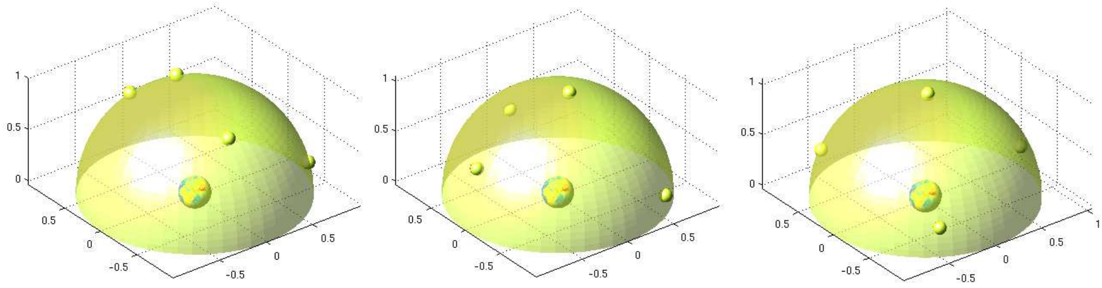


Figure 7: Input and output parameters and results for distribution of 45 satellites



Nr. of iterations: 5 GDOP: 4.0341 | Nr. of iterations: 50 GDOP: 1.8700 | Nr. of iterations: 500 GDOP: 1.7123

Figure 8: Progress of satellite distribution and values of respective GDOP coefficients

Derived calculation of GDOP coefficient was integrated into the computer program and value of GDOP can be computed for any setup of satellites. In order to show how this coefficient changes due to the satellites distribution is shown on the following example in *Figure 8*. In this case 4 satellites are being distributed evenly, the process is shown in pictures and respective values of computed GDOP are displayed bellow each step.

The computer program was tested for various numbers of satellites, different values of elevation angle over the horizon and for various numbers of iterations of genetic algorithm. *Table 1* shows input parameters and calculated DOP coefficients for three examples – 4, 12 and 45 satellites.

Table 1: Results and values of coefficients and input parameters for 4, 12 and 45 satellites

Nr. of satellites [-]	Angle above the horizon [º]	Nr. of iterations [-]	GDOP [-]	PDOP [-]	HDOP [-]	TDOP [-]
4	10	5,000	1.7322	1.6331	1.1547	0.3334
12	14	20,000	1.1460	1.0752	0.6537	0.1573
45	14	160,000	0.6139	0.5691	0.3471	0.0531

The model presented in this paper is considerably simplified. The Earth is considered as a regular sphere with no unevenness as mountains, buildings, trees, clouds etc. All these obstacles spawn deterioration in quality of satellite visibility and signal transmission. In reality it is possible to receive signal from 4 to 11 satellites at the time [5], thus values for 12 and 45 satellites shown in *Table 1* are not corresponding to reality, however it is shown that genetic algorithm is able to distribute even high amount of satellites above the hemisphere and that with more satellites used the better value of GDOP is obtained. The quality of DOP coefficients is evaluated according to the categorization shown in *Table 2* [6].

Table 2: DOP ratings

DOP	**Rating**
1	Ideal
2 – 3	Excellent
4 – 6	Good
7 – 8	Moderate
9 – 20	Fair
21 – 50	Poor

6. Conclusion

In the paper the derivation of GDOP coefficient and computed examples were shown. The calculations prove that coefficient is dependent on the distribution of satellites relatively to the user's position. More even the distribution is the better value (lower) of GDOP coefficient is obtained. Also more satellites are used for navigation the better value of GDOP is reached. The optimal distribution of satellites was reached using a genetic algorithm. Further work will be focused on better approximation of conditions that would approach a real situation. Also the possibility of use of neural networks for GDOP approximation will be examined.

Acknowledgement

The task was assigned and developed within the course of Modern systems of CNS (Control, Navigation and Surveillance) under the supervision of Prof. Ing. František Vejražka, CSc., professor of Department of Radioelectronics, Faculty of Electrical Engineering, CTU in Prague.

References:

[1] Dana, P.: *Global Positioning System Overview*, available on-line [March 9, 2011] <http://www.colorado.edu/geography/gcraft/notes/gps/gps.html>

[2] Zirari, S., Canalda, P., Spies, F.: *Modelling And Emulations Of An Extended GDOP For Hybrid And Combined Positioning System*

[3] Milliken, R.J., Zoller, C.J.: *Principle of Operation of NAVSTAR and System Characteristics*

[4] Majda, F.: *Využití umělé inteligence v operačním výzkumu*. Research project, FJFI CTU in Prague, 2009

[5] Bao, J., Tsui, Y.: *Fundamentals of Global Positioning System Receivers: A Software Approach*. John Wiley & Sons, 2000, electronic ISBN 0-471-20054-9

[6] Kaya, F., Saritas, M.: *A Computer Simulation of Dilution of Precision in the Global Positioning System Using Matlab*. Gazi University, Faculty of engineering and architecture