A. SÁRHIDAI
Eötvös Loránd Geophysical Institute
1145 Budapest, Columbus 17–23
(Hungary)

DESIGN OF FUNDAMENTAL GRAVITY NETWORKS BASED ON THE APPROXIMATION OF THE GIVEN VARIANCE–COVARIANCE MATRIX

Abstract

Techniques will be presented for the design of one-dimensional gravity nets by means of given variance-covariance matrices. After a critical review of the methods for the solution of the matrix equation $(\bar{A}^T \bar{P} \bar{A})^{-1} = \bar{Q}_x$, *we shall compare different numerical results in order to judge the quality of the designs carried out by means of an* SVD *criterion matrix, by a criterion matrix created according to an assumed distance-dependence of the mean errors of the grid points, and by means of an iteratively improved criterion matrix respectively.*

Introduction

The required accuracy and the amount of cost and labor involved in extended primary gravity networks make inevitable that the design of the networks be made by state-of-the-art techniques. The rapid development in computers provides new means for such computations that had rightly been thought intractable or uneconomic.

In the solution of the network design problem by means of given variance-covariance matrices – by the so-called criterion matrices – we are given the $\bar{A}$ matrix representing the measurements that can be carried out in the network, and the *a posteriori* variance-covariance matrix of the unknowns whose structure gets more and more closely approximated in the course of the design. We have to determine a generally diagonal matrix $\bar{P}$ which contains the weights of the measurements included in matrix $\bar{A}$. Or, more mathematically, in the case of fixed networks we have to solve the matrix equation

$$(\bar{A}^T \bar{P} \bar{A})^{-1} = \bar{Q}_x \tag{1}$$

while for free networks, the equation to be solved is

$$(\bar{A}^T \bar{P} \bar{A})^{+} = \bar{Q}_x \tag{2}$$

Bull. Géod. 60 (1986) pp. 355–376.

First we briefly review the different methods for the solution of Eqs. (1) and (2) and present three techniques for the determination of the criterion matrix of gravity networks. The results obtained by the different techniques will be compared through numerical examples.

1. Methods for the solution of the matrix equation $(\bar{A}^T \bar{P} \bar{A})^{-1} = \bar{Q}_x$

1.1. The direct solution U–solution

We express the matrix $\bar{P}$ from the equation

$$\bar{A}^T \bar{P} \bar{A} = \bar{Q}_x^{-1} \tag{3}$$

obtained upon inverting both sides of Eq. (1). If $\bar{P}$ is diagonal, that is, if the measurements are assumed as independent, the elements of $\bar{P}$ are obtained by the Khatri-Rao matrix multiplication

$$(\bar{A}^T \odot \bar{A}^T) \cdot \bar{p} = \bar{U} \cdot \bar{p} = \mathrm{vec}\,(\bar{Q}_x^{-1}) \tag{4}$$

where $\bar{p}$ is a vector formed from the unknown weights, vec $(\bar{Q}_x^{-1})$ is a vector derived from the upper triangle matrix of the inverse of the criterion matrix, that is,

$$\begin{aligned} \bar{p} &= (p_1\,,\ p_2\,, \ldots\ldots, p_m) \\ \mathrm{vec}\,(\bar{Q}_x^{-1}) &= (q_{11}, \ldots\ldots, q_{1n}\,, q_{22}\,, \ldots\ldots, q_{nn}) \end{aligned} \tag{5}$$

If the proposed measurements are not assumed as independent, i.e., if $\bar{p}$ is not a diagonal matrix, the elements of $\bar{p}$ can be expressed by a Kronecker product. The elements of vector $\bar{p}$ can be determined from the system of equations (4) by means of the relation

$$\bar{p} = \bar{U}^+ \cdot \mathrm{vec}\,(\bar{Q}_x^{-1}) \tag{6}$$

if the number of unknowns is greater than the number of equations; in the contrary case we use

$$\bar{p} = (\bar{U}^T \cdot \bar{U})^+ \cdot \bar{U}^T \cdot \mathrm{vec}\,(\bar{Q}_x^{-1}) \tag{7}$$

in the above equations $\bar{U}^+$ denotes the minimal-norm inverse to matrix $\bar{U}$. We realize from Eq. (7) that what we are really doing amounts to an approximation of the criterion matrix in the least mean square sense. The disadvantage of the method is that it can also yield negative weights that lack physical significance. According to my and others' (e.g., Sprinsky 1978) experiences, if we do get negative weights, the corresponding measurements must be omitted and the calculation carried out once more – this would lead to acceptable but very likely not optimal weights. (I don't know of any mathematical proof which excludes the possibility of further improvement of the target function, as one can see in mathematical programming.)

1.2. The canonical solutions

Following (B. Schaffrin, F. Krumm, D. Fritsch, 1980), we bring $\bar{Q}_x^{-1}$ into the form

$$\bar{Q}_x^{-1} = \bar{S}\,\bar{D}\,\bar{S}^T \tag{8}$$

where $\bar{S}$ consists of the eigenvectors of $\bar{Q}_x^{-1}$, matrix $\bar{D}$ is a diagonal matrix containing the eigenvalues of $\bar{Q}_x^{-1}$. Substituting Eq. (8) into (3), and since $\bar{S}^T\,\bar{A}^T\,\bar{P}\,\bar{A}\bar{S} = \bar{D}$ and $SDS^T = A^T\,PA$ rearrangement gives

$$(\bar{A}\bar{S})^T \odot (\bar{A}\bar{S})^T \cdot \bar{p} = \mathrm{vec}\,(\bar{Q}_x^{-1}) \tag{9}$$

It is an advantageous feature of the method that the eigenvalues of the variance-covariance matrix computed by means of the resulting measurement weights will be less than or equal to the eigenvalues of the criterion matrix. Consequently, the axes of the error ellipsoid of the designed network will also be less than or equal to the axes of the error ellipsoid belonging to the net defined by the criterion matrix.

1.3. The HR–solution

In order to eliminate the deficiencies of the negative weights, Wimmer (1981) elaborated an iterative method where the criterion matrix is also gradually approximated.

Let us bring the matrix equation $\bar{Q}_x = (\bar{A}^T\,\bar{P}\,\bar{A})^+$ to the form

$$\bar{Q}_x = \bar{H}\,\bar{P}^+\,\bar{H}^T \tag{10}$$

where we introduced the notation $\bar{H} = (\bar{A}^T\,\bar{P}\,\bar{A})^+\,\bar{A}^T\,\bar{P}$.

The first approximation of the weight vector $\bar{p}^+ = \mathrm{vec\,d}\,\bar{P}^+$ is provided from the solution of the system of equations

$$(\bar{H} \odot \bar{H}) \cdot \bar{p}^+ = \mathrm{vec}\,(\bar{Q}_x) \tag{11}$$

The solution is then similarly obtained as in case of the U–solution. The weight matrix is defined by the relation

$$\bar{p}^+ = \mathrm{vec\,d}\,\bar{P}^+ \tag{12}$$

where

$$\bar{p} = \begin{Bmatrix} 1/p_i^+ \\ \\ 0 \end{Bmatrix} \quad \text{if} \quad \begin{Bmatrix} p_i^+ \neq 0 \\ \\ p_i^+ = 0 \end{Bmatrix} \tag{13}$$

Since the weights are contained by matrix $\bar{H}$, the computational procedure must be repeated until the weights do not change any more. Usually, the first approximation of the $\bar{p}$ matrix is taken as the unit matrix.

Practical computations by Wimmer (1981) have shown that the optimal solution is fairly well approximated after five iterations. Numerically, the convergence to zero of the weights might cause difficulties. In such cases both too large and too small values of p_i^+ lead to the disappearance of p_i.

1.4. Solution of the problem by mathematical programming

A serious deficiency of all techniques dealt with thus far is that the occurrence of negative weights in the course of the solution cannot be excluded. In order to circumvent this problem, Grafarend (1975) proposed linear programming. By transforming Eq. (4) into the inequality

$$\bar{U} \cdot \bar{p} \leqq \bar{q} \qquad \bar{p} \geqq \bar{0} \tag{14}$$

and introducing the objective function

$$R_2 = \sum_{i=1}^{m} p_i \rightarrow max$$

we can determine weights which lead to the most accurate network, under the given conditions. On the other hand, the system of inequalities

$$\bar{U} \cdot \bar{p} \geqq \bar{q} \qquad \bar{p} \geqq 0 \tag{15}$$

and the objective function $R_2 = \sum_{i=1}^{m} p_i \rightarrow min$ lead to weights which can be used to design a minimum cost net under the given conditions.

Others (Cross and Thapa, 1979), in connection with the design of levelling networks by means of a criterion matrix, proposed the relations

$$\bar{U} \cdot \bar{p} \geqq \operatorname{vec}(\bar{Q}^{-1})_i \tag{16}$$

for the main diagonal of the variance-covariance matrix, and the inequalities

$$\bar{U} \cdot \bar{p} \leqq \operatorname{vec}(\bar{Q}^{-1})_j \tag{17}$$

for the off-diagonal elements, subject to the minimization of the objective function $\Sigma\, p_i \rightarrow min$.

In other words, they determined the designed measurement weights so as to ensure that the diagonal elements of the variance-covariance matrix of the resulting network should be less than or equal to the corresponding elements of the criterion matrix; and the off-diagonal elements should be greater than or equal to the corresponding elements of the criterion matrix. Consequently, the trace of the $\bar{Q}_x$ matrix of the designed network will be less than or equal to the trace of the criterion matrix while the mean errors of the connections after adjustment will be less than or equal to those designed.

Strictly speaking, linear programming can only be used in the case of continuous variables. In practical cases we are usually interested in finding repetition numbers for certain possible connections. As a matter of fact, linear programming does not generally provide integer solutions ! Making use of inequalities (14) we can restate our problem in terms of integer programming as

$$\begin{aligned} &\text{a)} \quad \bar{p} \in I_n^+ \\ &\text{b)} \quad \bar{U} \cdot \bar{p} \leqq \bar{q} \\ &\text{c)} \quad \bar{c}^T \cdot \bar{p} \rightarrow max \end{aligned} \tag{18}$$

where vector $\bar{c}$ consists of quantities proportional to the amount of labor required by a single execution of individual measurements. Even though the solution of the linear programming task that had been constructed in order to solve the matrix equation $\bar{A}^T \bar{P} \bar{A} = \bar{Q}_x^{-1}$ automatically assures positive weights, some of the measurement weights can still assume unreasonably large values because of the inequalities specified in the problem. This difficulty can be tackled if we prescribe further conditions, *viz.*, that the sum of squares of the differences between the elements of the criterion matrix and of the resulting variance-covariance matrix should be minimal.

That is, subject to the conditions (14) or (15) or (16) and (17), we introduce the objective function

$$[\bar{U} \cdot \bar{p} - \mathrm{vec}(\bar{Q}_x)]^T \cdot [\bar{U} \cdot \bar{p} - \mathrm{vec}(\bar{Q}_x)] \rightarrow min \tag{19}$$

The solution of this canonically given problem via quadratic programming contains all advantageous properties of the previously discussed solutions. If the problem does have a solution, the resulting weights are certainly nonnegative real values, the characteristic values of the corresponding variance-covariance matrix will be less than or equal to the characteristic values of the criterion matrix; the sum of squares of the differences between elements of the criterion matrix and of the variance-covariance matrix computed by the weights $\bar{p}$ will be minimal. The only drawback of the solution is that the obtained measurement weights are generally not integers.

The method starts out from decomposition (8) of the criterion matrix. With the notation $\bar{Z} = \bar{A} \cdot \bar{S}$ the problem to be solved becomes, in canonical form :

$$(\bar{Z}^T \odot \bar{Z}^T) \cdot \bar{p} = \mathrm{vec}(\bar{D}) \tag{20}$$

where vec $(\bar{D})$ is a vector constructed from the upper triangle matrix of the characteristic-value matrix $\bar{D}$.

For inconcistent systems the following quadratic programming problem has to be solved :

$$(\bar{Z}^T \boxdot \bar{Z}^T) \cdot \bar{p} \geqq \mathrm{vec}(\bar{D}) \tag{21}$$

$$[(\bar{Z}^T \odot \bar{Z}^T) \cdot \bar{p} - \mathrm{vec}(\bar{D})]^T \cdot [(\bar{Z}^T \odot \bar{Z}^T) \cdot \bar{p} - \mathrm{vec}(\bar{D})] = min \tag{22}$$

$$\bar{p} \geqq \bar{0} \tag{23}$$

where $\boxdot$ is the Kronecker product;

$$\bar{C} = \bar{A} \boxdot \bar{B} \text{ means } \qquad c_{ij} = a_{ij} \cdot b_{ij} .$$

2. Construction of the criterion matrix

One of the most crucial problems in second-order design is the definition of the criterion matrix.

In what follows we describe in detail the construction of the criterion matrix, under the assumption of distance-dependent mean errors of the network points, by transforming the characteristic values of the variance-covariance matrix, as well as by making use of iteratively improved criterion matrices.

2.1. Constructions of the criterion matrix of gravity networks in case of an assumed distance-dependent mean error of the network points.

For the design of gravity networks use can be made of the relationships derived for levelling nets (Cross and Thapa, 1979).

Assume that the variances of network points vary with interstation distance as

$$\delta_{G_{ij}} = f(\ell_{ij}) \tag{24}$$

where $\delta_{G_{ij}}$ is the variance of the difference between the adjusted gravity values at points i and j; ℓ_{ij} is the distance of the two points. The network should be designed so as to ensure that the variance computed from the gravity value differences after adjustment should be less than that computed from Eq. (24).

Suppose the variances of the network points vary as

$$\delta_{G_{ij}} = C \cdot \sqrt{\ell_{ij}} \quad , \text{ (where } C \text{ is constant)} \tag{25}$$

If the variance at the origin of the network is δ_s then, in terms of the distance between points i and S, the variance at point i becomes

$$\delta_i^2 = \delta_s^2 + f(\ell_{si}) \tag{25a}$$

$$\delta_{G_{ij}}^2 = \delta_i^2 + \delta_j^2 - 2 \cdot \delta_{ij} \tag{26}$$

where δ_i and δ_j are computed from Eq. (25a) the value of $\delta_{G_{ij}}$ comes from Eq. (25). With this procedure, the entire criterion matrix of the gravity network can be determined.

As an illustration of this method, consider the design of a net consisting of seven points. The first step of the design is to define the variance function. Because of the identical principles underlying gravimetric and levelling surveys – the quantity sought is determined from the difference of the values read off at two different stations – and as the reliability of the measurements in both cases depends on the distance between the stations, it seems reasonable to use a function of form (25) that has served reasonably well in case of levelling, that is, let

$$\delta_{ij} = 0.01\sqrt{\ell/100} \tag{27}$$

The mean error of the origin — taking into account the mean error of the absolute measurement — is taken to be 0.015 mgal. The variance of the variance-covariance matrix belonging to the origin is obtained upon substitution into the relationship

$$\mu_i = \mu_o\sqrt{Q_{ii}} \tag{28}$$

If we assume μ_o = 0.03 mgal (Csapó - Pollhammer - Sárhidai , 1981) , we get $Q_{ii} = Q_{33} = 0.25$. By making use of Eqs. (24) — (27) we similarly obtain the other elements of the criterion matrix. Next, we compute the coefficients of the equations for the adjustment of the measurable connections, construct the system of inequalities (16) and (17) and solve the resulting linear programming system subject to the condition $\Sigma\, p_i \rightarrow min$. The results obtained are compiled in *Table 1.* It is apparent from the *Table* that certain connections ought to be measured by very small weights in order to assure the required structure of the network. When these connections were left out and computations repeated, it was found that the weights of the remaining connections have but negligibly changed. This means the omitted measurements only affect the structure of the variance-covariance matrix i.e., the mean errors of the adjusted connections, they do not influence the mean errors of the adjusted points. The resulting weights, however, are very large. The design of the network on the basis of the variance function (27) is unrealistic taking into account obvious obstacles of instrumentation.

Let us repeat the design, using this time the variance function

$$\delta_{ij} = 0.02\cdot\sqrt{\ell/100} \tag{29}$$

As seen from *Table 1,* the obtained repetition numbers are already reasonable, the network can be economically realized within the possibilities of the available instrument park. From among the weights obtained in the first iteration, we omitted the smallest ones and repeated the computation. This restriction of the number of connections caused an increase in the weights for the connections 1–7 and 4–5 . In order to stay within the realm of actually realizable networks it is reasonable to limit the maximal distance for connections. If we carry out measurements only for the case of an assumed $\ell_{ij_{max}} \leqq 200$ km , the resulting weights do not significantly differ from those obtained in the previous examples *(Table 1).*

The computations were also carried out by means of integer programming. As seen from *Table 1,* the very small weights provided by linear programming becomes zeroes in integer programming. As for the other weights, they are the same as the rounded-up results of linear programming, except for the connections 1–7, 3–7, 4–7. The labor required to carry out the measurements can be further decreased, if we define the objective function $\Sigma\, p_i\, \ell_i$. As seen from the results summarized in *Table 1,* the weight of the connection between points 1 and 7 (being 200 km apart) has decreased from 3 to 1 , while for the pair 6–7 (80 km apart) the weight of the connection has increased from 3 to 5 . The saving in transportation is obvious.

The variance function (25) is only applicable for networks with one fixed point; such networks, however, are never designed in practice. In case of networks

Table 1
Repetition numbers of the test network, using a criterion matrix constructed under the assumption of fistance-dependent mean errors

Var. func. / target func.	$\mu_t=0.01\cdot\sqrt{\ell/100}$ $\Sigma p_i \to min$ 1. iteration	$\Sigma p_i \to min$ 2. iteration	$\Sigma p_i \ell_i \to min$	$\mu_t=0.02\cdot\sqrt{\ell/100}$ $\Sigma p_i \to min$ 1. iteration	$\Sigma p_i \to min$ 2. iteration	$\Sigma p_i \to min$ $t_{max} < 200$ km	$\Sigma p_i \to min$ $p_i \in I$	$\Sigma p_i \ell_i \to min$	$\Sigma p_i \ell_i \to min$ $p_i \in I$
1–2	5.74	5.74	5.74	1.435	1.44	1.44	2	1.43	2
1–3	4.00	4.00	4.00	1.00	1.00	1.00	2	1.00	2
1–4	0		0	0					
1–5	0.62		0.62	0.155					
1–6	0			0					
1–7	3.41	3.41	1.69	3.852	4.36	3.62	3	0.42	1
2–3	20.58	20.58	20.58	5.14	5.14	5.14	6	5.14	6
2–4	0			0					
2–5	0			0					
2–6	0.61		0.61	0.152		0.15	1	0.15	1
2–7	6.73	6.73	6.73	1.683	1.68	1.68	2	1.68	2
3–4	7.36	7.36	7.36	1.84	1.84				
3–5	0			0					
3–6	2.89	2.89	2.89	0.72	0.72	0.72	1	0.72	1
3–7	0			0		0.48	0	0.48	0
4–5	10.81	10.81	10.81	2.71	3.41	2.71	3	2.71	3
4–6	8.37	8.37	8.37	2.091	2.09	2.09	3	2.09	3
4–7	0			0		0.06	0	0.06	0
5–6	2.67	2.67	2.67	0.668	0.67	0.67	1	0.67	1
5–7	2.02	2.02	2.02	0.505		0.51	1	0.51	1
6–7	10.04	10.04	11.76	2.51	2.51	2.51	3	5.71	5

containing several fixed points, the interpretation of the distance ℓ_{ij} in Eq. (25) might be problematic.

An analysis of the errors of the Hungarian primary gravity network has proved that for networks containing several absolute measurements the mean errors of the network points do not show correlation with the distance measured from the absolute points, i.e., for such a gravity network the variance function (25) does not apply.

2.2. SVD criterion matrices

Sprinsky (1978) proposed a new way to construct criterion matrices, the method was further developed by Wimmer (1981). The main ideas can be interpreted as follows : let us accept some *a priori* matrix for the network to be designed (an obvious choice is $\bar{P} = \bar{E}$). Thus, the variance-covariance matrix of the network can be computed in the knowledge of the realizable measurements. Spectral decomposition of the matrix $\bar{Q}_x$ yields

$$\bar{Q}_x = \bar{S}\,\bar{D}\,\bar{S}^T \tag{30}$$

where $\bar{D} = \mathrm{diag}\,(\lambda_i)$ contains the eigenvalues of $\bar{Q}_x$, $\bar{S}$ is the matrix of the orthogonal eigenvectors. Geometrically, the matrix $\bar{Q}_x$ can be considered as an m-dimensional hyperellipsoid whose semiaxes are equal to the square roots of the eigenvalues, while the eigenvectors define the directions of the axes. The greater the accuracy of a network the less will be the length of the semiaxes of the error ellipsoid. The gist of Wimmer's method is that the criterion matrix $\tilde{Q}_x$ is determined so as to contract the spectra of matrix $\bar{Q}_x$ in such a manner that the eigenvectors are kept unchanged :

$$\tilde{Q}_x = \tilde{Q}_{SVD} = \bar{S}\,\tilde{D}\,\bar{S}^T \tag{31}$$

implying that

$$\mathrm{rank}\,(\tilde{Q}_{SVD}) = \mathrm{rank}\,(\bar{Q}_x) = k \tag{32}$$

$$\tilde{D} = \mathrm{diag}\,(\tilde{\lambda}_i) \leq \mathrm{diag}\,(\lambda_i) \tag{33}$$

$$\tilde{\lambda}_k = \lambda_k\,,\ \tilde{\lambda}_{k+1} = \tilde{\lambda}_m = 0$$

The eigenvalues $\tilde{\lambda}_i$ of matrix $\tilde{Q}_{SVD}$ are determined by means of a parameter t , by making use of the relationship

$$\tilde{\lambda}_i = min\,(\lambda_i\,,\tilde{\lambda}_1) \tag{34}$$

where

$$\tilde{\lambda}_1 = \lambda_1 - t\cdot(\lambda_1 - \lambda_k) \quad 0 \leq t \leq 1 \tag{35}$$

For $t = 0$ the eigenvalues remain the same; for $t = 1$ all eigenvalues become equal. The important advantage of the previously sketched procedure as compared with other methods to generate criterion matrices is that the defect of the criterion matrix does not change during transformation. Consequently, for free networks the defect of the criterion matrix $\tilde{Q}_{SVD}$ agrees with the defect of the original matrix. The definition of the defect of the criterion matrix, which, in case of criterion matrices of free networks is even now still partly unsolved (Sárközy, 1980), does not require any special computation.

2.2.1. Application of SVD criterion matrices for the design of gravity networks

In order to test the applicability of SVD criterion matrices, and the other procedures dealt with in Section 1 we carried out the design of a test network consisting of 7 points. The design was carried out both with and without the incorporation of absolute measurements. For each algorithm, its efficiency and the quality of the resulting network were checked for the values $t = 0.1$, 0.3, 0.6 and 0.9 of the parameter. For the results obtained by means of the U–technique *(Table 2)* (where $\mu_G = G\left(\frac{\text{spur}\,(\bar{Q}_x)}{k}, \det\,(\bar{Q}_x)^{1/k}\right)$ is the general Gaussian error, s_Q is the standard deviation of the diagonal elements of the $\bar{Q}_x$ matrix), it can be stated that for fixed networks the parameters characterizing the network only slightly change for values between $t = 0.1 - 0.7$; the only significant change occurs for $t = 0.9$. The parameters of the resulting variance-covariance matrix, however (λ_{max}, trace, etc.) lag behind the corresponding parameters of the criterion matrix. This is due to the fact that in the course of design only relative measurements had been taken into account. In order to grant that the net should meet the level of accuracy required by the criterion matrix, the reliability (or weight) of the absolute measurements and of the "tie-in" measurements between the airport points should be increased. In case of the U–solution, and the solution obtained by quadratic programming, only slight changes occur the structure of the weights of the measurements in case of weights p_2, p_{13} and p_{15} – for the transformation parameter $t = 0.9$. The two solutions do not differ too much; that obtained via quadratic programming is somewhat better.

If we solve the problem by means of linear programming, the structure of the weights would significantly change. The network built up from the resulting weights, however, will still be much inferior than designed. The result – as far as the parameters of the variance-covariance matrix are concerned – is worse than in case of the U–solution or quadratic programming. This unfavourable result is mostly due to the behaviour of the variance-covariance matrix as a response to the transformation of the greatest characteristic value. When the problem is tackled by linear programming, generally no solution is obtained unless inequalities (16) and (17) are both prescribed. The systems of conditions (14) (high accuracy) and (15) (minimal cost) only infrequently lead to a solvable problem. If we start out from a system of inequalities of canonical form, this does not have a solution, either. This means that this special kind of solution gives us "mathematical hints" in course of the calculations, warning us that no variance-covariance matrix of the required form can be established by changing the weights of the relative measurements.

As for the design of free networks by means of SVD criterion matrices, we

Table 2

Fixed network

Parameters	U-solution t = 0.5	U-solution t = 0.9	Quadratic programming t = 0.5	Quadratic programming t = 0.9
Q_{11}	0.470	0.455	0.469	0.450
Q_{22}	0.566	0.551	0.564	0.541
Q_{33}	0.674	0.571	0.671	0.559
Q_{44}	0.610	0.592	0.607	0.577
Q_{55}	0.499	0.482	0.498	0.477
Q_{66}	0.583	0.568	0.581	0.556
Q_{77}	0.485	0.471	0.484	0.465
spur	3.887	3.690	3.847	3.625
det	0.000358	0.000217	0.000344	0.000168
cond	15.833	16.034	15.877	16.436
spect	2.225	2.210	2.225	2.206
λ_{max}	2.375	2.357	2.374	2.348
Q_{mean}	0.555	0.527	0.553	0.518
s_Q	0.075	0.056	0.074	0.047
μ_G	0.431	0.405	0.429	0.395
p_{1-2}	1.006	1.009	1.011	1.039
p_{1-3}	0.007	0.401	0.011	0.439
p_{1-4}	1.003	1.012	1.009	1.044
p_{1-5}	0.997	0.986	1.000	1.007
p_{1-6}	1.004	0.989	1.008	1.008
p_{1-7}	0.004	0.014	0.009	0.046
p_{2-3}	0.993	1.291	1.003	1.358
p_{2-4}	0.004	0.021	0.011	0.063
p_{2-5}	1.006	1.009	1.011	1.039
p_{2-6}	1.009	1.026	1.016	1.069
p_{2-7}	1.001	1.007	1.006	1.039
p_{3-4}	1.012	1.264	1.022	1.339
p_{3-5}	0.008	0.396	0.014	0.436
p_{3-6}	1.016	1.263	1.028	1.339
p_{3-7}	0.012	0.365	0.014	0.408
p_{4-5}	1.026	1.052	1.042	1.142
p_{4-6}	1.002	1.022	1.010	1.067
p_{4-7}	1.010	1.021	1.015	1.055
p_{5-6}	1.006	1.016	1.011	1.049
p_{5-7}	0	0	0.003	0.011
p_{6-7}	1.009	1.022	1.015	1.056

obtained more favourable results *(Table 3)*. Identical results were obtained in the course of the U–solution, and for the solution of the canonically posed problem by means of quadratic programming respectively. The eigenvalues of the variance-covariance matrix belonging to the solution are practically identical with those of the criterion matrix. For $t = 0.9$ both the eigenvalues and the mean errors of the points become identical, i.e., an ideal state has been achieved ! The weight of the individual measurements is 1.712, uniformly for all connections. The same result is obtained if the solution is sought by linear programming. The repetition numbers obtained by integer programming are the same as the results of linear programming, rounded up to the next integer.

The realizability of gravity networks is greatly influenced by the distance between the points of measurement or, more exactly, by the time lapsed between subsequent measurements. Keeping this in mind, the design has been carried out once more, limiting this time the maximal distance of the connections to 200 km .

Naturally, in the variant where all connections are assumed as measurable, the parameters characterizing the network will be better than in the network designed by means of connections of limited length.

Let us now consider the sum of the necessary repetition numbers. For free networks, the sum is 22 in the interval $t = 0 - 0.5$, for $t = 0.7$ it is 26 , while for $t = 0.9$ it is already 65 . In the latter case we have not succeeded in attaining the ideal state that was achieved when all measurements had been used. Anyway, this variant is obviously uneconomical because of the jump-like increase in repetititon number. *(Figure 1a, 1b.)*

For fixed networks the repetition numbers barely change with increasing values of the parameter t . In the interval $t = 0.1 - 0.7$ the total number of measurements is 20 , even for $t = 0.9$ it is only 22 . The insensitivity of the SVD criterion matrices against the changes in the parameter t is also proved by the fact that for $t = 0$ – i.e., when only the original network, built up by the incorporation of all measurable connections must be substituted by sides of $\ell_{max} \leq 200$ km length – the number of measurements obtained from integer programming was 23 , that is, greater than in the cases obtained upon transforming the eigenvalues *(Figure 2)*.

The sketches of the free networks, as computed by different parameters t , as well as the parameters of these networks are shown in ***Fig. 1a*** and ***Fig. 1b*** for fixed network; the results are shown for the cases $t = 0.1$ and $t = 0.9$ in ***Fig. 2a*** and ***Fig. 2b***.

From the point of view of numerical analysis it might be interesting that for fixed networks for any single value of the parameter t we get 3–5 solutions consisting of integer weights. As the transformation of the maximal eigenvalue has not proved successful for fixed networks, we tried to construct the criterion matrix by means of reduging ***all*** eigenvalues by the same ratio :

$$\tilde{\lambda}_i = t \cdot \lambda_i \tag{36}$$

Using this criterion matrix, the systems of inequalities (16) and (17) were solved by linear programming. Comparing these results to those obtained by transforming the maximal eigenvalue alone, it turns out how strongly the values of the trace, of the determinant and of λ_{max} decrease with decreasing values of the parameter t . The

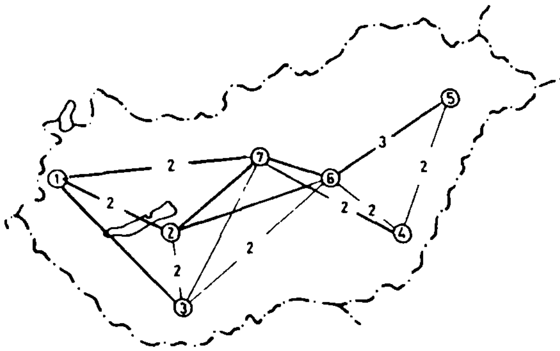
Table 3

Linear and integer programming

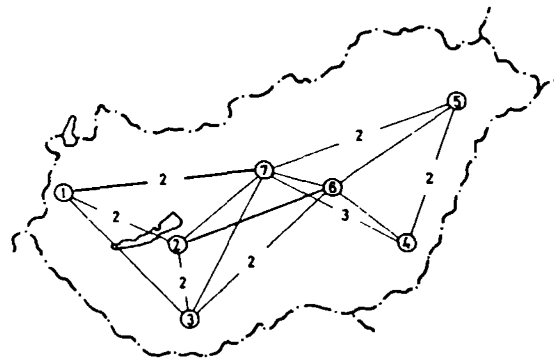
Free network

Parameters	t =0.1		t =0.3		t =0.5		t =0.7	t =0.9
	cont	integer	cont	integer	cont	integer	cont	cont
Q_{11}	0.208	0.175	0.199	0.176	0.190	0.176	0.181	0.072
Q_{22}	0.180	0.137	0.178	0.161	0.175	0.161	0.171	0.072
Q_{33}	0.636	0.229	0.495	0.229	0.355	0.229	0.214	0.072
Q_{44}	0.209	0.144	0.208	0.188	0.206	0.188	0.192	0.072
Q_{55}	0.244	0.157	0.235	0.207	0.225	0.207	0.197	0.072
Q_{66}	0.178	0.137	0.176	0.161	0.174	0.161	0.171	0.072
Q_{77}	0.206	0.175	0.199	0.176	0.192	0.176	0.179	0.072
spur	1.861	1.154	1.690	1.298	1.517	1.298	1.305	0.500
det	0.000356	0.000039	0.000273	0.000088	0.000190	0.000088		
cond	4.665	2.246	3.627	1.849	2.590	1.849		
spekt	–	–	–	–	–	–	–	–
λ_{max}	0.751	0.274	0.584	0.281	0.417	0.281	0.250	0.834
Q_{mean}	0.266	0.165	0.241	0.185	0.217	0.185	0.186	0.072
s_Q	0.165	0.033	0.114	0.025	0.064	0.025	0.016	0
μ_G	0.288	0.188	0.268	0.214	0.246	0.214		
p_{1-2}	0.999	1	0.989	1	0.971	1	0.934	1.712
p_{1-3}	0.074	1	0.285	1	0.666	1	1.556	4.755
p_{1-4}	0.997	1	0.989	1	0.975	1	0.933	1.712
p_{1-5}	0.996	1	0.980	1	0.952	1	0.880	1.712
p_{1-6}	0.990	1	0.973	1	0.941	1	0.879	1.712
p_{1-7}	0	0	0	0	0	0	0	1.712
p_{2-3}	1.041	2	1.156	2	1.362	2	1.826	4.755
p_{2-4}	0	0	0	0	0	0	0.046	1.712
p_{2-5}	1.000	2	0.991	1	0.973	1	0.822	1.712
p_{2-6}	0.998	1	0.994	1	0.987	1	0.970	1.712
p_{2-7}	0.994	1	0.986	1	0.970	1	1.006	1.712
p_{3-4}	1.035	2	1.126	2	1.291	2	1.717	4.755
p_{3-5}	0.070	1	0.276	1	0.647	1	1.456	4.755
p_{3-6}	1.029	2	1.115	2	1.270	2	1.632	4.755
p_{3-7}	0.063	1	0.243	1	0.567	1	1.359	4.755
p_{4-5}	1.007	2	0.986	1	0.948	1	1.517	4.755
p_{4-6}	1.002	2	0.999	1	0.993	1	0.979	1.712
p_{4-7}	0.999	1	0.992	1	0.980	1	0.801	1.712
p_{5-6}	0.996	1	0.988	1	0.976	1	0.946	1.712
p_{5-7}	0	0	0	0	0	0	0.096	1.712
p_{6-7}	0.996	1	0.990	1	0.979	1	0.952	1.712

Integer programming free network
($t_{max} < 200$ km)

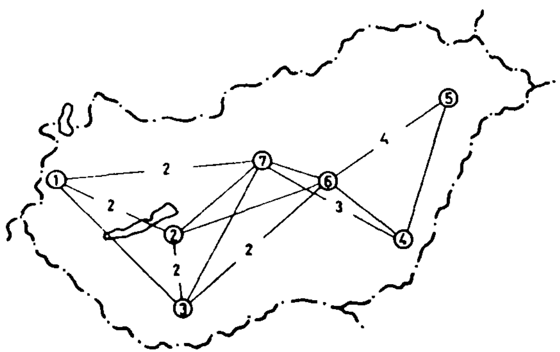


t=0, t=0.1



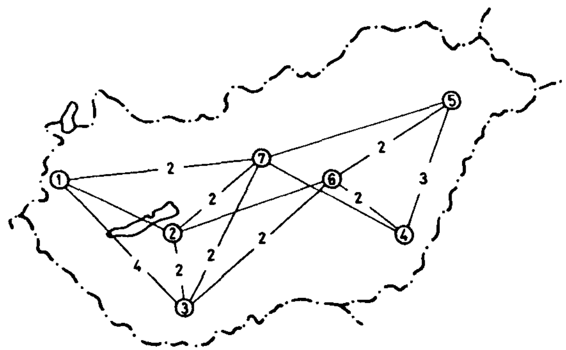
t=0.5 2. version

Parameters t	Solution with different values of t		
	0	0.3	0.5
n	22	22	22
spur	1.512	1.525	1.437
det	0.000084	0.000082	0.000077
cond	4.534	5.376	5.581
λ_{max}	0.535	0.500	0.480
$\overline{Q_{ii}}$	0.216	0.218	0.205
s_Q	0.083	0.080	0.082
μ_G	0.230	0.231	0.223

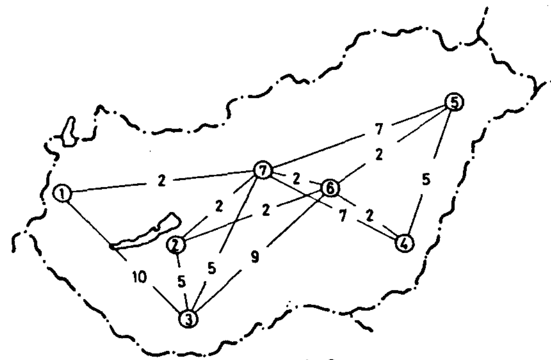


t=0.5
1.version

Figure 1 a.

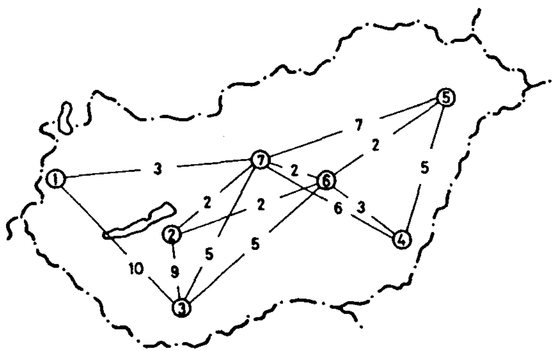


t=0.7



t=0.9

2.version



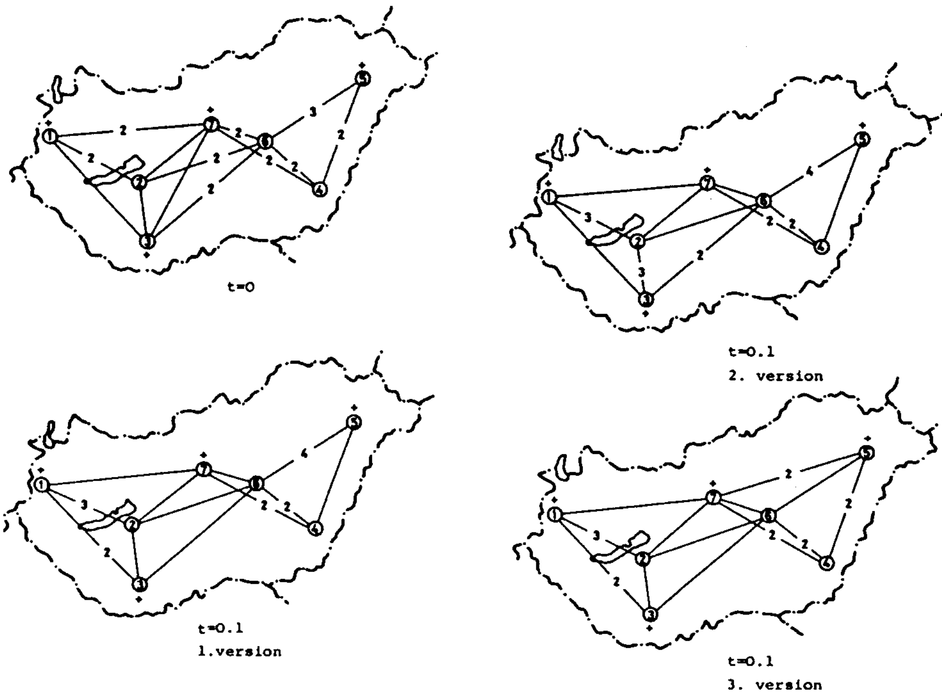
t=0.9

1.version

Parameters t	Solutions with different values of t		
	t=0.7	t=0.9 1.version	t=0.9 2.version
n	26	65	65
spur	1.298	o.533	o.543
det	o.oooo45	2.2o4·1o^{-7}	2.342·1o^{-7}
cond	4.594	5.297	5.592
λ_{max}	o.487	o.196	o.2o3
$\overline{Q}_{ii}$	o.185	o.o76	o.o78
s_Q	o.o47	o.o17	o.o17
μ_G	o.2o2	o.o83	o.o84

Figure 1 b.

Fixed network
(t_{max} < 200 km)

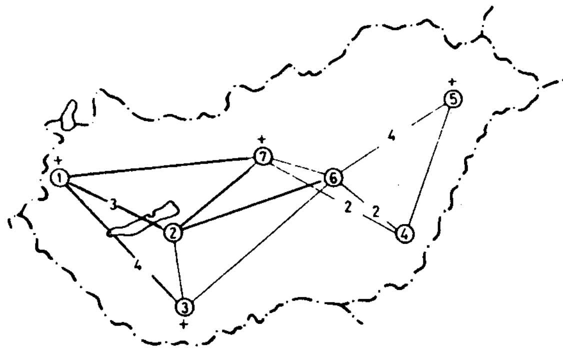


Parameters	Solution			
	t=0	t=0.1 1.version	t=0.1 2.version	t=0.1 3.version
spur	3.547	3.805	3.722	3.781
det	0.000005	0.000141	0.000140	0.000138
cond	29.213	27.395	21.518	22.686
spect	2.257	2.270	2.254	2.277
λ_{max}	2.337	2.356	2.339	2.382
$\bar{Q}_{ii}$	0.507	0.544	0.539	0.540
s_Q	0.059	0.062	0.058	0.070
μ_G	0.368	0.402	0.400	0.400

Legends:

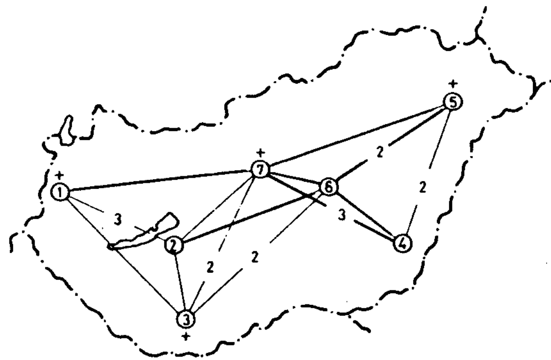
— 4 — repetition number
(4) station with relative measurements
\+ absolute measurement

Figure 2a.

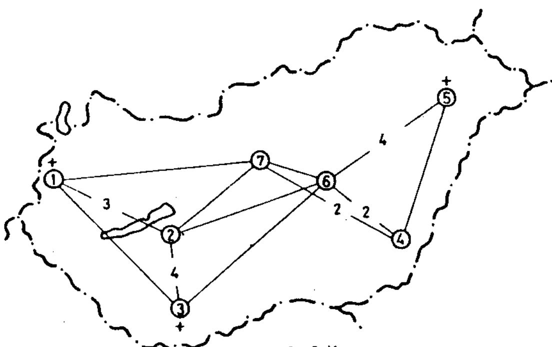


t=0.9/a

Fixed network
(t_{max} < 200 km)



t=0.9/c



t=0.9/b

Parame- ters	Solutions for the same problem		
	a	b	c
spur	3.7o4	3.679	3.597
det	o.oooloo	o.oooo99	o.oooo99
cond	27.326	27.o58	24.2o8
spekt	2.264	2.241	2.228
λ_{max}	2.35o	2.327	2.324
$\overline{Q}_{ii}$	o.529	o.526	o.514
s_Q	o.o6o	o.o57	o.o53
μ_G	o.388	o.386	o.38o

Figure 2b.

decrease of parameter s_Q in the case of $t = 0.1$ deserves special attention : the Q_{ii} values of the network points are practically the same, at least for sufficiently high repetition numbers *(Table 4).* On the other hand, the rapid increase of the condition number of the network is an important warning : the improvement in network parameters is counterbalanced by a serious distortion of the error ellipsoid of the network ! It is interesting that for free networks the weights of measurements can be obtained from the weights of the initial network from the relation

$$p_i^t = p_i / t \tag{37}$$

where t is the actual value of the parameter. *(Table 4).*

Summing up the different techniques of solution it can be stated that the U–solution is simple, requires little computer time and, its results have favourable properties. Generally, one can recommend such algorithms for the solution where the squared deviations between the elements of the criterion matrix and of the resulting variance-covariance matrix are to be taken into account, since in this way we can avoid the occurrence of unrealistically large weights. From among all solutions, the U–solution is least time consuming. For 21 variables the CPU time on our RYAD 35 (which has about the same parameters as an IBM 145) stays below 10 s, even in the case when negative weights are iteratively omitted. The time required for linear programming (for 21 variables and inequalities) is 15–20 s; for the algorithms based on quadratic programming, it becomes 1–2 minutes. Most time consuming is integer programming (branch and bound algorithm) : for 14 variables it takes 8–10 min., for 21 variables the CPU time reaches 70–75 minutes !

2.3. Iterative improvement of the criterion matrix

It frequently occurs in practice that it is found out only after a network had already been established that the mean error of certain of its point deviates from the other ones in a nonallowable manner. In such cases, the straighforward solution is to directly change the elements of the variance-covariance matrix (Boedecker 1979). *Figure 3* shows a network, obtained by simulation, where the mean error of point 3 is significantly greater than that of the other points. Let us change the mean error of this point such that it should be equal to the average value of the mean errors of the other points. The covariances corresponding to this point can also be assumed in a first approximation to be the average of the covariances for the other points. After the first iteration step the trace decreases by some 20% , the determinant by more than 90%, the value of s_Q by 30%. The value λ_{max} has remained practically the same and the condition number increased. The number of necessary measurements has increased at the same time from 15 to 27 . After the second iteration the trace decreases by a further 10%, the value of parameter s_Q by 60%, the required number of measurements becomes 43 instead of 27 .

Conclusions

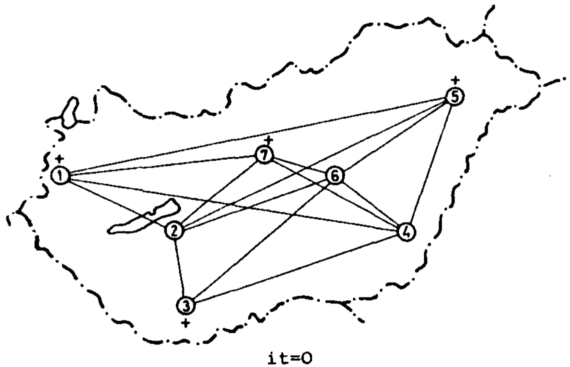
Analysing the numerical solutions it proved many times that rounding up the continuous solution to the next integer we get larger repetition numbers of measurements than if we solved the problem by integer programming. Having constructed the criterion matrix of gravity networks in case of an assumed distance – dependent

Table 4

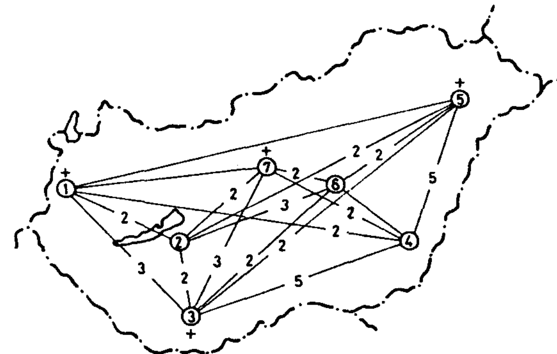
Linear programming, transforming the eigenvalues of the criterion matrix by $\tilde{\lambda}_i = t \cdot \lambda_i$

Parameters	Fixed network		Free network	
	t =0.1	t =0.9	t =0.1	t =0.9
Q_{11}	0.326	0.452	0.021	0.191
Q_{22}	0.338	0.543	0.018	0.164
Q_{33}	0.346	0.637	0.071	0.635
Q_{44}	0.342	0.582	0.021	0.189
Q_{55}	0.331	0.481	0.025	0.224
Q_{66}	0.340	0.558	0.018	0.161
Q_{77}	0.329	0.468	0.021	0.188
spur	2.352	3.721	0.195	1.752
det	$2.965.10^{-10}$	0.000193	–	–
cond	147.00	17.323	–	–
spect	2.190	2.220	–	–
λ_{max}	2.205	2.356	0.083	0.751
Q_{mean}	0.336	0.532	0.028	0.250
s_Q	0.007	0.068	0.019	0.171
μ_G	0.153	0.404	–	–
p_{1-2}	10.013	1.113	10.020	1.113
p_{1-3}	20.011	0.748	0	0
p_{1-4}	9.987	1.11	9.995	1.111
p_{1-5}	9.933	1.104	10.009	1.112
p_{1-6}	10.013	1.113	9.965	1.107
p_{1-7}	0	0	0.007	0.001
p_{2-3}	9.826	1.092	10.004	1.112
p_{2-4}	0	0	0	0
p_{2-5}	10.006	1.112	10.037	1.115
p_{2-6}	10.037	1.115	9.995	1.111
p_{2-7}	9.956	1.106	9.968	1.108
p_{3-4}	10.003	1.111	10.025	1.114
p_{3-5}	0.035	0.004	0	0
p_{3-6}	10.040	1.115	9.991	1.110
p_{3-7}	0.065	0.007	0.004	0
p_{4-5}	10.125	1.125	10.148	1.128
p_{4-6}	9.972	1.108	10.028	1.114
p_{4-7}	10.048	1.116	10.011	1.112
p_{5-6}	10.009	1.112	9.981	1.109
p_{5-7}	7.168	0.085	0	0
p_{6-7}	10.045	1.116	9.983	1.109

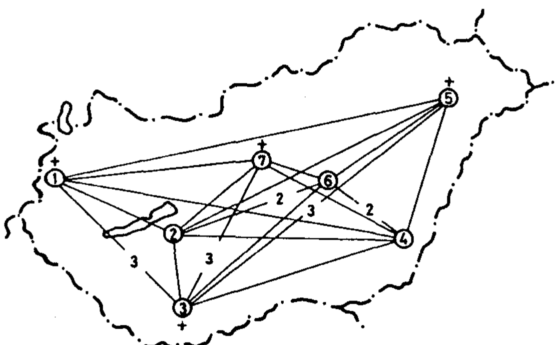
Iterative improvement of the network
(fixed network)



it=0



it=2



it=1

Para-meters	Number of iterations		
	0	1	2
spur	3.900	3.300	2.879
det	0.000376	0.000030	0.000002
cond	15.742	21.098	30.093
spekt	2.226	2.251	2.182
λ_{max}	2.377	2.363	2.257
$\overline{Q}_{ii}$	0.557	0.471	0.411
s_Q	0.076	0.049	0.018
μ_G	0.433	0.337	0.267

Figure 3.

mean error of the network points, the repetition numbers of measurements by integer programming were every time smaller than those rounded up to the next integer in the continuous solution. *(Table 1.)*

So networks with few points should be computed by integer programming. For larger networks which contain more than 10 points, integer programming is uneconomic. The most suefficient algorithm of integer programming is the branch and bound method since it has more favourable numerical features than the cutting plane procedure.

The parameters of the resulting variance-covariance matrix lag behind the corresponding parameters of the SVD criterion matrix in the case of a fixed network. We were unable to approach the criterion matrix by any of the solution methods. This was due to the fact that in the course of design only relative measurements had been taken into account. In order to grant that the net should meet the level of accuracy required by the criterion matrix, the reliability and the number of the absolute measurements and of the "tie-in" measurements between the airport points should be increased. So, the SVD criterion matrices make too strict terms for a fixed network.

From the point of view of numerical analysis it might be interesting that for fixed networks for any single value of the parameter t we get 3–5 solutions consisting of integer weights. It means that we can get same value of the target function by different network configurations.

In the course of designing free networks by means of an SVD criterion matrix we succeeded in finding a measurement arrangement which exactly corresponded to that required by the criterion matrix.

By comparing the results obtained from the different solutions it can be stated that for free networks the U–solution, the solution obtained by linear programming and the convex quadratic-programming solution of the problem posed in canonical form have all led to the same results.

In case of reducing all eigenvalues by (36), the weights of measurements for free networks can be obtained from the weights of the initial network by a linear relation. So, in this case it is unnecessary to solve the matrix equation (1).

The weights in *Table 4* characterize the accuracy of results which were obtained in the course of linear programming. Deviations between exact and computed results are about 5%.

o

o o

BIBLIOGRAPHY

G. BOEDECKER : Development of an observation scheme from the variance-covariance matrix of the unknowns, in Optimization of design and computation of control network. Proceeding of IAG Symp. in Sopron, 1979.

P.A. CROSS and K. THAPA : The optimal design of levelling networks. Survey Review, 1979. XXV. (192), pp. 68–79.

G. CSAPÓ, Mné POLLHAMMER, A. SÁRHIDAI : Processing and adjustment of the measurements in 1978–79 on the Carpathian-Balkan polygon. (In Hungarian). SzOGSz Rept., Budapest ELGI Archives, 1981.

E. GRAFAREND : Second order design of geodetic nets. ZfV. 1975/4.

F. SÁRKÖZY : Optimale Plannung von Ingenieurfestpunktnetzen. Periodica Politechnika, Budapest, 1980. pp. 1–2.

G. SCHMITT : Experiences with the second-order design problem in theoretical and practical geodetic networks. IAG Symposium. Sopron. 1977.

B. SCHAFFRIN, F. KRUMM, D. FRITSCH : Positiv–diagonale Genauigkeitsoptimierung von Realnetzen über den Komplementaritaets-Algoritmus. VIII. Int. Kurs. für Ingenieuervermessung, Zürich, 1980.

W.H. SPRINSKY : Improvement of parameter accuracy by choice and quality of observation. Bull. Géod. 52, pp. 269–279. 1978.

H. WIMMER : Ein Betrag zur Gewichtsoptimierung geodätischer Netze. DGK Rep. C. 269, Munich, 1981.

Received : 28.02.1986
Accepted : 12.11.1986