

TRANSFORMING THE L1-NORM ADJUSTMENT OF
A LEVELING NETWORK INTO A FLOW PROBLEM

by
W.-D. Schuh

Institute of Theoretical Geodesy,
Technical University Graz,
Rechbauerstraße 12
A-8010 Graz, AUSTRIA

7 th International Symposium on Geodetic Computations. Presented Paper.
Cracow, Poland, June 18-21, 1985.

W.-D. Schuh
Institute of Theoretical Geodesy,
Technical University Graz
A-8010 Graz, AUSTRIA

TRANSFORMING THE L1-NORM ADJUSTMENT OF A LEVELING NETWORK INTO A FLOW PROBLEM

Summary

For a leveling network, the adjustment by minimizing the absolute sum of residuals can be transformed by the aid of the linear program method into a flow problem. Thereby the advantage is gained that the investigation of the stability and of checking possibilities for the solution is simple. This is shown for two different methods by means of a small test example. Another important advantage of the proposed method is the very fast computation and the low storage requirements. It is shown that newly developed flow algorithms are highly superior to the conventional L1-norm computation procedures used up to now.

1. Introduction

Due to the development of automatic data flows from measurements to self-contained evaluation by a computer, the importance of automatic error recognition methods and of outlier tests is strongly growing. During the last years, the adjustment by minimizing the absolute sum of residuals (L1-norm adjustment) has been developed as a robust procedure for the determination of erroneous measurements. Fuchs (1980) investigated this procedure for general geodetic networks. A specialization to leveling networks with equal weights has been performed by Meissl (1980) and by Fuchs, Hofmann-Wellenhof, Schuh (1983) resulting in a simple algorithm, saving computer time and storage space.

In this paper, the original problem is re-formulated in terms of a new representation where the adjustment by minimizing the absolute sum of residuals for arbitrary leveling networks is transformed into a capacity circulation flow problem. Using the rules of graph- and flow theory (Tarjan (1983), Dantzig (1966)), new aspects related to the stability with respect to outliers and related to checking possibilities are found. Moreover, the very efficient algorithms of flow theory can be used for the solution of this flow problem (cf. Domschke (1981), Murty (1984)).

2. Problem formulation

Since we use various methods of graph theory, we start with explaining some of its features. We consider a finite leveling network of arbitrary shape with m points of unknown heights H_i . In order to determine these heights, n height differences $h_{i,j} = H_j - H_i$ with weights $p_{i,j}$ (repetition numbers) are measured. This configuration may be represented by a graph. The points with unknown heights are called nodes and are comprised within the set V .

Remark: In the following sections, we consider only free networks. The modifications necessary for fixed points are shown in sec. 7.

The measured height differences $h_{i,j}$ are denoted as arcs (i,j) and are comprised within the set E . The network (or leveling network) is completely determined by the graph $G=(V,E)$. In case of a contiguous network, $m-1$ observations enable a unique relative height determination

of the nodes for the whole network. These observations are comprised within the set S where S is called "spanning tree".

The problem for the adjustment of leveling networks by minimizing the absolute sum of residuals can be formulated by the target function

$$\sum_{(i,j) \in E} p_{ij} |v_{ij}| \dots \text{Min} , \quad (1)$$

which is to be minimized where the residuals v_{ij} are computed by

$$h_{ij} + v_{ij} = H_j - H_i . \quad (2)$$

Splitting up the residuals as differences of non-negative numbers

$$v_{ij} = v_{ij}^+ - v_{ij}^- \quad \text{where} \quad v_{ij}^+, v_{ij}^- \geq 0 \quad (3)$$

then the target function (1) may be rewritten as

$$\sum_{(i,j) \in E} p_{ij} v_{ij}^+ + \sum_{(i,j) \in E} p_{ij} v_{ij}^- \dots \text{Min} . \quad (4)$$

Now the observation equations (2) are given by

$$H_j - H_i - v_{ij}^+ + v_{ij}^- = h_{ij} , \quad (5)$$

with restrictions

$$H_j, H_i \in R \quad R \dots \text{real numbers} \quad (6)$$

$$v_{ij}^+, v_{ij}^- \geq 0 \quad (7)$$

The formulae mentioned above, together with the linear target function (4), the conditions (5), and the restrictions (6) and (7), respectively, constitute a linear program. A general representation is given by

$$c^T U + d^T V \dots \text{Min} \quad (8)$$

where

$$AU + BV = W \quad (9)$$

and

$$U \in R \quad (10)$$

$$V \geq 0 \quad (11)$$

For the solution of this problem, various algorithms of linear programming can be used. A general but very time- and storage consuming solution is possible by the Simplex procedure (Dantzig (1966)) and its different variations. Taking into account the special structure of inconsistent equations, Barrodale and Roberts (1973) show an acceleration of the procedure (the necessary memory requirement is in the order of $n*m$). The application of those general methods to geodetic networks is shown e.g. in Fuchs (1980), Schmid (1980), Ebong (1985).

Looking at the coefficients of the observation equation (5), it can be seen that there is a sparse pattern of values -1 and $+1$. Thus the use of graph theory and the similarity with networks of linear programming

is evident. Related to this, Meissl (1980) shows a solution procedure for equally weighted leveling networks. This method is extended by Fuchs et al (1983) and compared with other adjustment procedures with respect to the robustness for finding out erroneous observations.

Since the structure of the observation equations (i.e., matrix A in (9)) is very important for the subsequent considerations, we show the following example.

Example: nodes 1,2,3,4 with unknown nodal values H_1, H_2, H_3, H_4

arcs	arc values	weights
(1,2)	26	8
(1,3)	12	4
(1,4)	28	2
(2,3)	-19	4
(2,4)	5	3
(3,4)	20	1

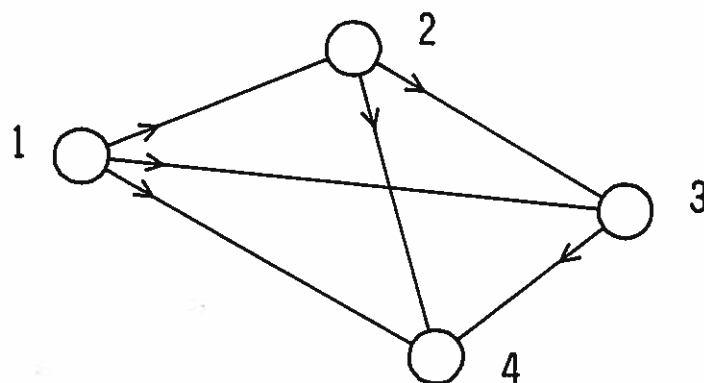


Figure 1. Simple leveling network

Observation equations:

H_1	H_2	H_3	H_4	v_{12}^+	v_{13}^+	v_{14}^+	v_{23}^+	v_{24}^+	v_{34}^+	v_{12}^-	v_{13}^-	v_{14}^-	v_{23}^-	v_{24}^-	v_{34}^-	$h_{1,j}$
-1	1			-1						1						26
-1		1			-1						1					12
-1			1			-1						1				28
	-1	1					-1						1			-19
	-1		1					-1						1		5
		-1	1						-1						1	20

Target function:

$$8v_{12}^+ + 4v_{13}^+ + 2v_{14}^+ + 4v_{23}^+ + 3v_{24}^+ + v_{34}^+ + 8v_{12}^- + 4v_{13}^- + 2v_{14}^- + 4v_{23}^- + 3v_{24}^- + v_{34}^- \dots \text{Min}$$

Combining the coefficients belonging to the unknown heights to matrix A, the unknown heights to the vector H, the observations to h, the weights to p, the residuals to v^+ and v^- , then the observation equations are obtained by

$$AH - Iv^+ + Iv^- = h \quad (12)$$

where I represents the unit matrix. Now the target function reads

$$0H + p^T v^+ + p^T v^- \dots \text{Min} \quad (13)$$

where 0 represents the zero matrix, and where we have the restrictions

$$v^+, v^- \geq 0 \quad \text{and } H \in R \quad (14)$$

Considering the rank of the observation equation (12) (free network,

n equations with $2n+m$ unknowns), we get from the unit matrix n linearly independent columns. In a contiguous network, matrix A contains m-1 linearly independent columns, since there is no information on the absolute height. It is known from the theory of linear programming that an optimal solution based on

$$v_{ij}^+ \geq 0 \implies v_{ij}^- = 0 \quad (15)$$

$$v_{ij}^- \geq 0 \implies v_{ij}^+ = 0 \quad (16)$$

yields a basis with n columns and m-1 non-basis variables. The values of all non-basis variables must be zero in order to minimize the target function.

By a mere formalism (Tucker scheme), the dual problem can be formulated for this linear program (formulae (12) through (14)), cf. Fuchs (1980), p.15; Dantzig (1966), p.145. This dual problem with n auxiliary variables x_{ij} (each x_{ij} adjoined to one observation) which are comprised within the vector x reads

$$A^T x = 0 \quad (17)$$

$$\left. \begin{array}{l} -x \leq p \\ x \leq p \end{array} \right\} \quad -p \leq x \leq p \quad (18)$$

$$h^T x \dots \text{Max} \quad (19)$$

Eq. (17) is obtained from columns 1 through m of the observation equations if one takes into account that the unknown nodal values

H_1, \dots, H_m may attain arbitrary real values (corresponding to the equality sign) and the cost factors for H are zeroes in the target function (13). Inequalities (18) are derived from columns $m+1$ through $m+n$ and $m+n+1$ through $m+2n$, respectively, of the observation equations with the corresponding cost factors p and the inequality sign because of restriction (14). The new target function (19), which is to be maximized, is obtained by the right-hand side of eq. (12).

Problem (17) through (19) represents again a linear program with the target function (19) and conditions (17), (18). By switching to the dual form for this dual problem, the original (primal) problem is obtained. Looking at condition (17) and taking into account the structure of a leveling network, these m equations can, in a simplified way, be represented by

$$-\sum_{k \in P(i)} x_{ki} + \sum_{j \in S(i)} x_{ij} = 0 \quad \text{für } i=1, \dots, m, \quad (20)$$

where $P(i)$ contains all neighbouring predecessors of node i , and $S(i)$ denotes all neighbouring successors of node i . Interpreting the auxiliary variable x_{ij} as unknown flow values, then condition (20) may be denoted as a flow conservation condition at each node. Using indexed notation, eq. (18)

$$-p_{ij} \leq x_{ij} \leq p_{ij} \quad (21)$$

gives capacity bounds for the flow values x_{ij} . The target function

$$\sum_{(i,j) \in E} h_{ij} x_{ij} \dots \text{Max} \quad (22)$$

is to be maximized. This is equivalent to the minimization of the target function for negative cost factors

$$\sum_{(i,j) \in E} -h_{ij} x_{ij} \dots \text{Min} \quad (23)$$

By the target function (23), the flow conservation condition (20) and the capacity bounds (21), a linear program is formulated which is known as capacity circulation flow problem.

Using procedures that are economic in terms of computer time and storage, this problem can be calculated very fast. The memory space needed is a function of the order $(n+m)$. The number of computation steps may be reduced to $(n*m \log n)$ depending on the algorithm (Domschke (1981), Tarjan (1983)), cf. sec. 8.

If the optimal solution of the dual problem is known, then the optimal solution of the original primal problem can be immediately specified, due to the orthogonality properties between the primal and the dual problem (complementary slackness condition). These orthogonality conditions cause the following correlation of two stations

$$x_{ij} = -p_{ij} \quad ==> \quad H_j - H_i \geq h_{ij} \quad (24)$$

$$x_{ij} = p_{ij} \quad ==> \quad H_j - H_i \leq h_{ij} \quad (25)$$

$$-p_{ij} \leq x_{ij} \leq p_{ij} \quad ==> \quad H_j - H_i = h_{ij} \quad (26)$$

On solving linear programs, some algorithms use the primal problem as well as the dual problem for the determination of the optimal solution. A specific algorithm using this primal-dual solution method for networks is the so-called 'Out-of-Kilter' algorithm. The basic elements of this algorithm are described in the subsequent section.

3. First method of solution

For the primal-dual methods of solution, first an arbitrary feasible solution is determined for the primal problem. Afterwards the solution of the dual problem is computed using the orthogonality condition of optimal solutions. If the dual problem is also feasible, i.e., all conditions and restrictions must be fulfilled, then the calculated solution is optimal due to specific properties of linear programming.

This procedure will now be explained by means of the example mentioned in sec. 2. Since the primal and dual problem formulations are equivalent (the dual problem of the dual problem leads back to the primal problem), we use the capacity flow problem (formulae (20), (21) and (23)) as the primal task formulation. We must start with the determination of a feasible flow for the leveling network. Since the zero-flow (x_{ij} for all $(i,j) \in E$) is a feasible solution, this flow is chosen as initial flow. As an initial solution of the dual problem, the values of all unknown heights are chosen as zero. Fig. 2 shows the initial situation where the preliminary values of the heights at the

nodes are written within the corresponding circles. Along the arc, the observation, the weight, and the flow are shown.

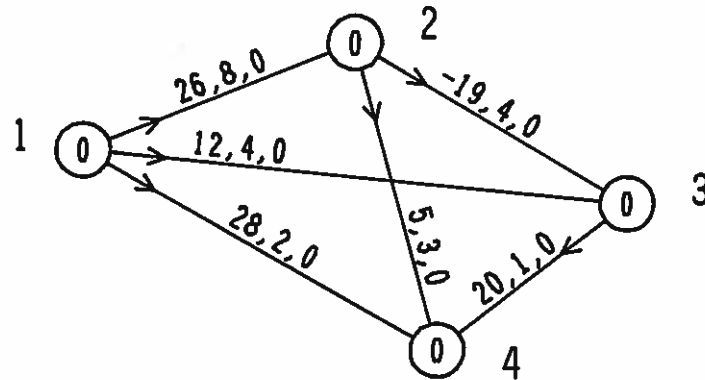


Figure 2. Initial situation

This primal problem is feasible, since the flow conservation condition (20) is fulfilled at each node and, moreover, the flow does not exceed the given bounds (21). This solution would be optimal, if conditions (24) through (26) are fulfilled for all arcs. Since arc (1,2) does not satisfy condition (26), it must be tried to change the situation in such a way that this condition is fulfilled. There are two alternatives. Either the flow of the arc is fitted to the upper bound where flow condition (20) must be maintained, then condition (25) will be fulfilled, or the nodal value is changed, so that condition (26) is fulfilled. We first choose the latter possibility and increase the value of the unknown node 2 to 26. Now condition (26) is fulfilled for arc (1,2) regardless of the flow, therefore the flow can be arbitrarily chosen in the range of the upper and the lower bound by the weight. This possibility is indicated in Fig. 3 by a bold line.

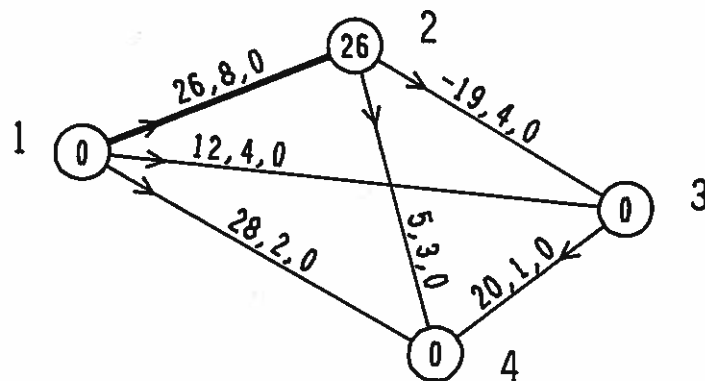


Figure 3. Arc (1,2) fulfills condition (26)

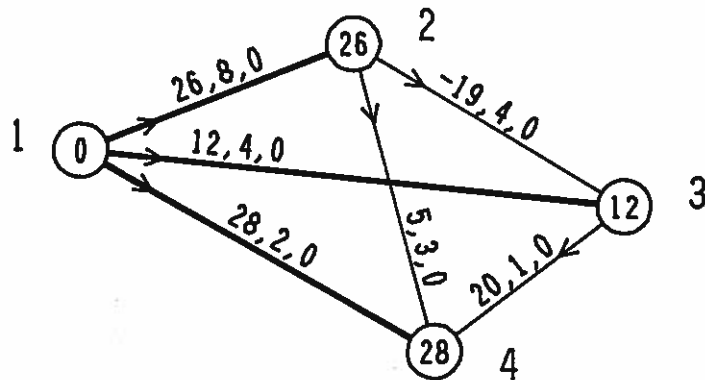


Figure 4. Arcs (1,2), (1,3) and (1,4) fulfill condition (26)

In an analogous way arcs (1,3) and (1,4) are treated, leading to the situation of Fig. 4. Now all arcs of node 1 fulfill the orthogonality conditions and form feasible solutions in both linear programs. Investigating now the arcs linked to node 2, we see that arc (2,3) violates the conditions: inserting the values for the unknowns H_1 , H_2 and the observations h_{1j} yields $12 - 16 \geq -19$. Using the preliminary unknowns, the orthogonality condition can be fulfilled if the flow of the arc is equal to the lower bound (condition (24)). This arc fulfills the orthogonality condition, if the flow can be distributed over the

remaining arcs (arcs shown by bold lines and arcs yet untouched). From now on, the flow of arc (2,3) must not be changed any more.

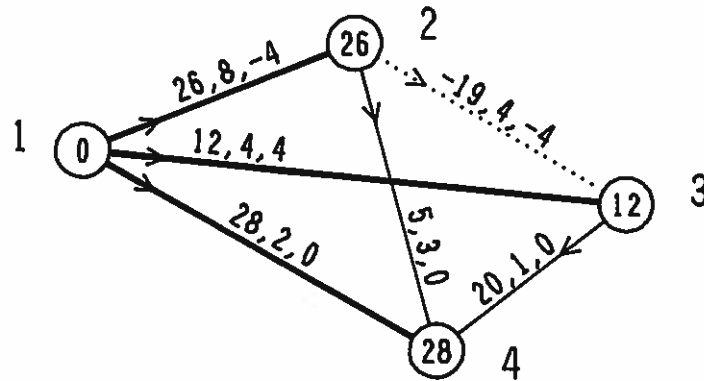


Figure 5. Arc (2,3) fulfills condition (24) after flow correction

As it can be seen from Fig. 5, three different kinds of arcs exist:

- * Arcs determining an unknown height provisionally (bold lines). The flow within the bounds is arbitrary because $H_j - H_i = h_{ij}$ is fulfilled.
- * Arcs between nodes which do not satisfy the condition $H_j - H_i = h_{ij}$ (dotted lines). The flow of these arcs is determined by the relation of condition (24) and condition (25).
- * Arcs not treated yet.

On investigating arc (2,4), again a contradiction occurs, because condition $H_4 - H_2 = 5$ is not fulfilled. To bring this arc to an optimal status, it would be necessary to increase the flow by 3. Since this flow cannot be distributed over the changeable arcs (the distribution over arcs (1,4), (2,4) would be possible but the remaining flow at node 3 cannot be put into (2,3), since (2,3) must not be changed and arc (1,3) may not get an additional positive flow), the value of the

unknown node 4 must be changed to 31 (fulfillment of condition (26)).

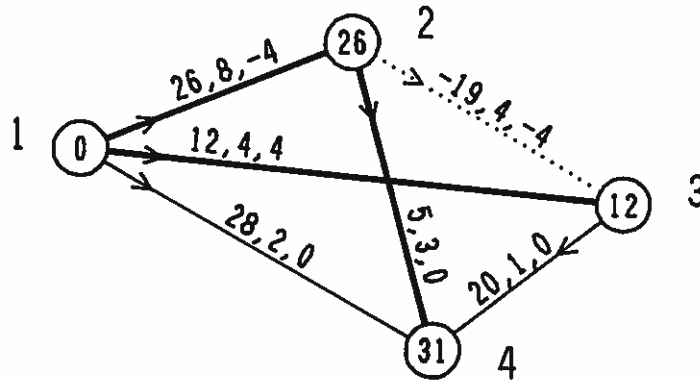


Figure 6. Arc (2,4) fulfills condition (26)

Arc (1,4) therefore does no more fulfill the orthogonality condition. But this can be corrected by a decrease of the flow in the loop formed by the nodes 4,2,1,4.

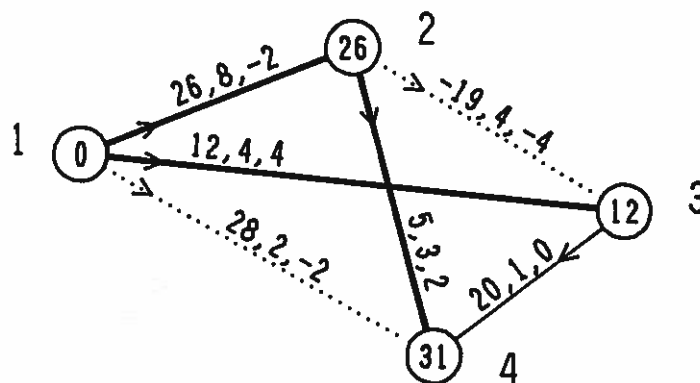


Figure 7. Correction of flow, arc (1,4) fulfills condition (24)

The only arc still violating conditions (24) through (26) is now (3,4). Since we have $31-12 \leq 20$, a change of the flow by 1 would be necessary. Since this flow cannot be transported from 3 to 1 due to the upper bound, the unknown node value at node 3 must be changed in such a way

that arc (3,4) fulfills condition (26).

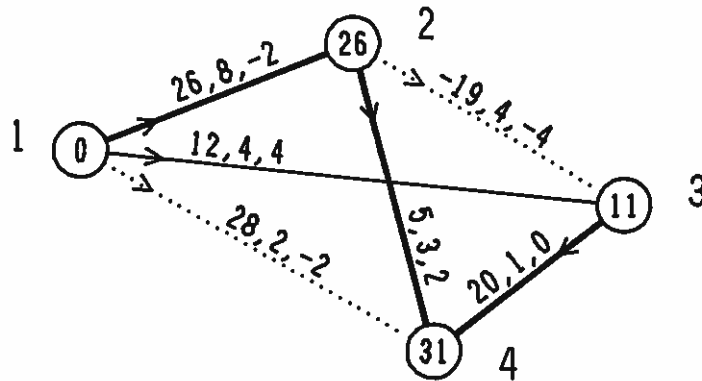


Figure 8. Optimal solution

Since now all arcs fulfill the primal as well as the dual problem, and no violation appears with respect to the orthogonality relations, the obtained solution is optimal. The purpose of the given example is to give an idea of the solution procedure and to enable an application of the method for very small networks. For large networks, existing flow algorithms using very sophisticated data structures may be used, cf. Domschke (1981) und Murty (1984).

4. Second method of solution

For the first method of solution, a feasible primal problem (flow problem) was set up, the results have been transformed into the dual problem using the orthogonality conditions, and a check was made for approval. Now, for the second method of solution, which represents an extension of Meissl's method (1980) to arbitrary weights, one starts with a feasible dual solution, transforms the unknowns into the primal

problem and tests for approval. By changing the unknown nodal values and the flow values, a feasible solution in the primal problem is searched for (fulfillment of the capacity bounds for the flows). The procedure is shown with the aid of the simple example of sec. 2.

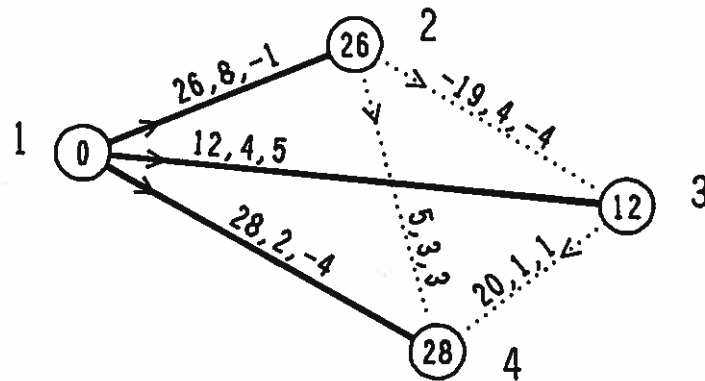


Figure 9. Initial situation

The procedure is started with an arbitrary choice of a spanning tree (in our example (1,2), (1,3) und (1,4), cf. Fig. 9) and the computation of the unknown nodal values (the reason for this can be found in sec. 2). Due to the orthogonality conditions

$$H_j - H_i \geq h_{ij} \implies x_{ij} = -p_{ij} \quad (27)$$

$$H_j - H_i \leq h_{ij} \implies x_{ij} = p_{ij} \quad (28)$$

the flows of the remaining arcs can be obtained. Using flow conservation conditions, the flow of the spanning tree may be determined (beginning with a node containing only one arc of the

spanning tree). If all arcs fulfill the capacity bounds

$$-p_{ij} \leq x_{ij} \leq p_{ij} \quad , \quad (29)$$

then an optimal solution is found. Otherwise a non-feasible arc must be eliminated from the tree, thereby two partial trees are generated, cf. Fig. 10.

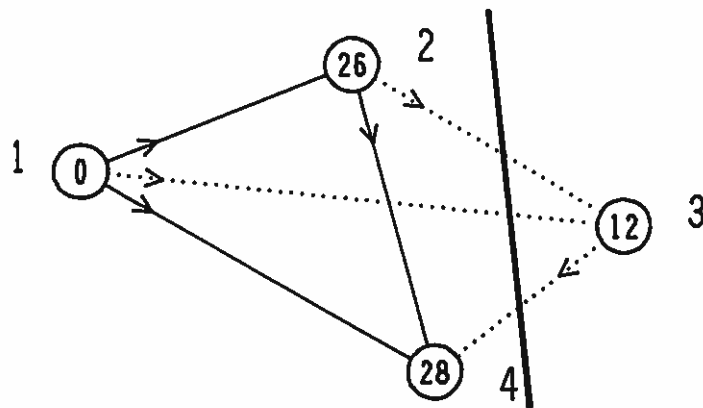


Figure 10. Arc (1,3) eliminated from the tree generates the subtree 1,2,4 and the subtree 3

Now all arcs leading from one subtree to the other are investigated. If the weighted median for the residuals of all those arcs is taken into the spanning tree, then a feasible solution of the primal and the dual problem is produced. Related to our example, the arcs (1,3), (2,3), (3,4) with residuals and weights 0,4; 5,4; and -4,1 are the corresponding arcs. If the observation from 3 to 4 is transformed into the form 4 to 3, i.e., directing to the second subtree, then the weighted median of the values 0,4; 5,4; and 4,1, is taken. This has the consequence that arc (1,3) of the spanning tree is replaced by (3,4). The flow is recomputed according to the procedure illustrated above.

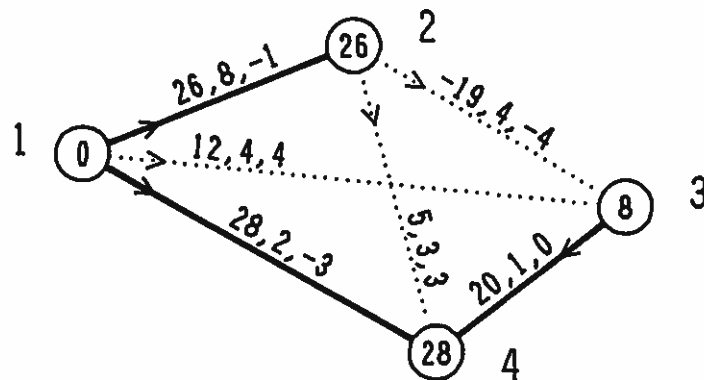


Figure 11. New spanning tree (1,2),(1,4) and (3,4)

Eliminating the non-feasible arc (1,4) from the tree and taking arc (2,4), which corresponds to the median value, into the spanning tree, we get the solution shown in Fig. 12.

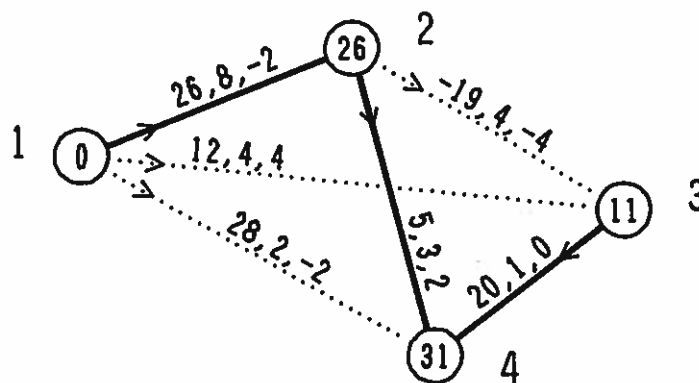


Figure 12. Optimal solution by the spanning tree (1,2),(2,4),(3,4)

Since this solution is feasible for the primal problem as well as for the dual one, an optimal solution is found.

5. Stability properties of the optimal solution

Considering the obtained optimal solution, two groups of observations can be discerned. The first group is formed by those arcs whose flow value is the upper or lower bound and which fulfill either condition (24) or condition (25). These arcs are called saturated arcs. Only the signs of the residuals for these observations are relevant which must satisfy either

$$H_j - H_i \geq h_{ij} \quad \text{for} \quad x_{ij} = -p_{ij} \quad (30)$$

or

$$H_j - H_i \leq h_{ij} \quad \text{for} \quad x_{ij} = p_{ij} \quad (31)$$

The value of the residual, obtained in the usual way by

$$v_{ij} = H_j - H_i - h_{ij} \quad (32)$$

is of no consequence with respect to the adjustment. Therefore the size of the residual has no influence on the obtained result. As a consequence, the residuals of erroneous observations are exactly reproduced. Observations containing a 10 meter error will obtain a residual of 10 meters.

The second group of observations is generated by arcs fulfilling condition (26).

$$H_j - H_i = h_{ij} \quad \text{for} \quad -p_{ij} \leq x_{ij} \leq p_{ij} \quad (33)$$

If special cases (ambiguous solutions) are not considered, then there exist $m-1$ observations of this type. This is a result of sec. 2 based on the rank of the adjustment procedure (non-basis variables). These arcs satisfy the condition $-p_{ij} \leq x_{ij} \leq p_{ij}$. As it can be easily found for the initial problem, these $m-1$ arcs are linearly independent. Therefore they form a spanning tree in the graph. In addition it follows that the non-saturated arcs do not permit a circuit, since otherwise a flow would be generated. This contradicts the optimality of the solution. All arcs whose capacity cannot be exhausted must satisfy the condition $H_j - H_i = h_{ij}$. They determine the obtained solution. We can make the following statements on the stability of the adjustment, if the degree of a node is determined by the sum of all possible capacities of those arcs leading to and coming from the node.

1. $p_{ij} > \text{degree}/2$

The capacity of this observation cannot be fully exhausted, since a distribution of the flow over the other arcs is not possible. This brings about that the observation, independent of its observation value, is always in the spanning tree and influences therefore the solution.

2. $|p_{ij}| > |x_{ij}|$

The weight of these arcs could be decreased or increased as long as the condition above remains valid without changing the result.

3. $|p_{ij}| = |x_{ij}|$

If the residual is not equal to zero, then only the sign but not

the size has an influence on the result. For a result equal to zero, an ambiguous solution occurs. In case of ambiguity, a change of the observation changes the unknowns but not the sum of the target function.

6. Reliability

Since observations with large weights are taken for the solution regardless of their observation values, e.g., arc (1,2) of the example mentioned, we describe a method, how such observations can be checked. This is necessary, because observations with large weights can also contain gross errors, e.g. errors due to wrong writing or point confusion. Considering for the second proposal of solution the way, how an observation is taken into the spanning tree (weighted median of all arcs between subtree 1 and subtree 2), we see that by means of the optimal solution any observation of the spanning tree may be checked. In order to do this, one must remove arc by arc and investigate the change of the median value. This enables, regardless of the choice of weights, an estimation of the influence of any observation on the adjusted network. As it can be seen for this method, the observations have only a local influence.

7. Fixed points

Up to now we have only investigated networks with unknown absolute height. This means that no fixed points were considered. The introduction of a fixed height can be achieved by an additional arc to

an auxiliary point where the capacity is restricted in that sense that it is greater than half of the degree of the corresponding point. If there are more fixed points available in the leveling network, then for each fixed point an arc to the one auxiliary point with the corresponding capacity bounds must be introduced. This causes, due to the considerations mentioned above, that this arc is in any case taken into the spanning tree. Thereby the height difference between the fixed points and the auxiliary point corresponds exactly to the arc value. If zero is chosen for the height of the auxiliary point and if this height value is maintained during the computation procedure (if this is not possible, a reduction of all nodes after the computation must be performed in such a way that the auxiliary point attains again the value zero), then we have reached the desired goal of fixing these nodal values.

8. Comparing the computation time for conventional methods and the Out-of-Kilter algorithm

Some networks have been computed using the Out-of-Kilter algorithm realized by Domschke (1981), pp.177-184, and using the conventional Barrodale and Roberts (1973) algorithm. The calculations were performed on a VAX 11/725. We quote as an example a leveling network of the Bundesamt für Eich- und Vermessungswesen in Vienna, Austria, which has been computed by two variants. In the first case, all inner points of the leveling chains were eliminated and only points with at least 3 links to other points were introduced into the adjustment. This network consists of 91 nodes and 121 arcs, where 1 node must be fixed. The

Out-of-Kilter algorithm takes 27 sec for the solution of this problem without needing approximate heights and 15 sec if using approximate values (measurement accuracy 0.0001 m, height differences greater than 10 m). The method of Barrodale and Roberts takes 3 min, no matter if approximate values are available or not.

The same network has been computed a second time without eliminating the inner points. This network consists of 186 nodes and 216 arcs. For the Out-of-Kilter algorithm, the solution takes 80 sec without and 40 sec with approximate values, whereas for the Barrodale-Roberts solution, 23 min are needed.

9. Conclusion

The first part of this paper deals with the adjustment by minimizing the absolute sum of residuals applied to leveling networks. This problem can be transformed into a flow problem by the aid of known methods of linear programming. This leads to statements and illustrations on the stability and reliability of the solution. In order to show the methods of solution, two methods have been explained by means of a small example which can be calculated by hand. Only additions and subtractions are necessary. For the solution of larger problems, sophisticated flow algorithms are available which are superior to the methods used up to now for the L1-solution of leveling networks. The behaviour with respect to computation time is excellent for these algorithms especially in case of weakly overdetermined systems and loop-like shapes. The algorithm can be applied to any

approximate solution. The number of digits to be gained has a logarithmic influence on the computation time. The storage space needed is a function which linearly increases with the number of unknowns and observations $(m+n)$, whereas the method of Barrodale and Roberts needs a storage location in the order of $(m*n)$.

Finally it is mentioned that during recent times some fast flow algorithms have been developed where the computational expenditure decreases enormously, cf. Ahrens and Finke (1980).

10. References

- Ahrens, J.H.; Finke, G. (1980): Primal transportation and transshipment algorithms. Physica Verlag, Würzburg. Operations Research, vol. 24, pp.11-32.
- Barrodale, I.; Roberts, F. (1973): An improved algorithm for discrete L1 linear approximation. SIAM J.Num.Ana., Vol.10, No.5.
- Collatz, L.; Wetterling, W. (1971): Optimierungsaufgaben. Springer-Verlag, Berlin-Heidelberg-New York, 2.Edition.
- Dantzig, G.B. (1966): Lineare Programmierung und Erweiterungen. Springer-Verlag, Berlin-Heidelberg-New York.
- Domschke, W. (1981): Logistik: Transport. Oldenbourg Verlag, München-Wien.
- Ebong, M. (1985): The least sum adjustment of a geodetic levelling network. Manuscripta geodaetica 10, pp.32-36.
- Fuchs H. (1980): Untersuchungen zur Ausgleichung durch Minimierung der Absolutsumme der Verbesserungen. Dissertation, Technical University Graz.

- Fuchs, H.; Hofmann-Wellenhof, B; Schuh, W.D. (1983): Adjustment and gross error detection of levelling networks. In: H.Pelzer and W.Niemeier (Editors): Precise Levelling. Dümmler Verlag, Bonn, pp.391-409.
- Meissl, P. (1980): Adjustment of leveling networks by minimizing the sum of absolute residuals. Proceedings of the Second International Symposium on Problems Related to the Redefinition of the North American Geodetic Network, Ottawa, Canada, pp.393-415.
- Murty, K. (1984): Linear and combinatorial programming. John Wiley and Sons, New York-Chichester-Brisbane-Toronto.
- Schmid, H. (1980): Die Bündelausgleichung nach der 1-Norm. Diploma work. Technical University Graz.
- Tarjan, R.E. (1983): Data structures and network algorithms. Society for Industrial and Applied Mathematics. Regional Conference Series in applied mathematics, 44, Philadelphia.