

Power Flow with Load Uncertainty¹

L.V. BARBOZA², G.P. DIMURO³, R.H.S. REISER⁴, Universidade Católica de Pelotas, 96010-000 Pelotas, RS, Brazil.

Abstract. This paper presents a methodology to solve load flow problems in which the load data are uncertain due to measurement errors. In order to deal with those uncertainties we apply techniques of Interval Mathematics. The algorithm uses the Interval Newton's method to solve the nonlinear system of equations generated by the problem. The implementation was performed in the Matlab[®] environment using the Intlab toolbox. In order to assess the performance of the proposed algorithm, the method was applied to hypothetical electric systems. In this paper, we present results for a three-bus network.

1. Introduction

Load flow [5] calculations provide power flows and voltages for a specified power system subject to the regulating capability of generators, condensers and tap changing under load transformers as well as specified net interchange between individual operating systems. This information is essential for the continuous evaluation of the current performance of a power system and for analyzing the effectiveness of alternative plans for system expansion to meet increased load demand. These analyses require the calculation of numerous load flow for both normal and emergency operating conditions.

The load flow problem consists of the calculation of power flows and voltages of a network for specified bus conditions. Associated with each bus are four quantities: the real and reactive power, the voltage magnitude and the phase angle. Three types of buses are represented in the load flow calculation and two of the four quantities mentioned above should be specified at every bus. It is necessary to select one bus, called the slack bus ($V\theta$), to provide the additional real and reactive power to supply the transmission losses, since these are unknown until the final solution is obtained. At this bus the voltage magnitude and phase angle are specified. The remaining buses of the system are designated either as voltage controlled buses (PV) or load buses (PQ). The real power and voltage magnitude are specified at PV buses. The real and reactive power are specified at PQ buses.

The mathematical formulation of the load flow problem results in a system of algebraic nonlinear equations. These equations can be established by using the bus

¹This work is partially supported by the Brazilian funding agency CTINFO/CNPq.

²Escola de Engenharia e Arquitetura, e-mail: luciano@ucpel.tche.br

³Escola de Informática, e-mail: liz@ucpel.tche.br

⁴Escola de Informática, e-mail: reiser@ucpel.tche.br

analysis that results in voltages as independent variables. Thus, the admittance network matrix should be used.

The solution of the system of algebraic equations that describes the power system is based on an iterative technique [11] because of the nonlinearity of its equations. This solution must satisfy Kirchhoff's laws, that is, the algebraic sum of all flows at a bus must be equal to zero and the algebraic sum of all voltages in a loop must also be equal to zero. We use the former law as a test for convergence of the solution in the iterative computational method.

All loads in the electric system are provided by measurement instruments, which frequently are inaccurate. Moreover, the specified variables, like real power at *PV* buses, also can be uncertain since their values are obtained via measurement equipment. This uncertainty in the input data can be enlarged due to both rounding and truncating processes that occur in numerical computation. As a consequence the actual error presented in the final results can not be easily evaluated. In order to rigorously control and automatically handle these numerical errors we propose to apply techniques of Interval Mathematics [7, 8, 9] for a more reliable load modelling.

Interval Mathematics considers a set of methods for handling intervals that approximate (contain) uncertain actual data. These methods are based on the definition of both interval arithmetic and optimal scalar product. Besides, the maximal accuracy principle guarantees (by means of the directed rounding) the automatic control of errors in numerical computation.

The main considerations of our approach in the development of an effective and reliable computer methodology in order to solve the above problem are: (i) the formulation of a mathematical description of the problem considering an interval modelling; (ii) the application of a numerical interval method for the solution of the system; (iii) the interrelation between (i) and (ii).

In this work we consider the application of this methodology to a hypothetical system composed by three buses. The solution of the nonlinear system provided by the problem is obtained by the Interval Newton's algorithm. The implementation was performed in Matlab[®] environment, using Intlab toolbox developed by S. Rump [10].

This paper is organized as follows. Firstly, in Section 2. we state the interval load flow problem. Next, Section 3. presents the interval Newton's method. The algorithm we propose to solve the problem is introduced in Section 4. An application of this methodology to a hypothetical three-bus system is discussed in Section 5. Finally, the conclusions are presented in Section 6.

2. Interval Load Flow Problem

The equation describing the performance of the network of a power system using the bus frame of reference in admittance form is

$$\mathbf{I} = \mathbf{Y}_{bus}\mathbf{E}, \quad (2.1)$$

where \mathbf{I} is a vector related to the current injection at the system buses, \mathbf{Y}_{bus} is the admittance matrix and \mathbf{E} is a vector with the complex nodal voltages. From (2.1),

it is possible to write

$$I_p = \sum_{q=1}^n Y_{pq} E_q. \quad (2.2)$$

The real and reactive power at any bus p is

$$P_p - jQ_p = E_p^* I_p. \quad (2.3)$$

This load flow problem can be solved by the Newton's method using a set of nonlinear equations to express real and reactive powers in terms of bus voltages [4, 12, 14]. Substituting I_p from equation (2.2) into equation (2.3) results in

$$P_p - jQ_p = E_p^* \sum_{q=1}^n Y_{pq} E_q. \quad (2.4)$$

Using cartesian coordinates, we have $E_p = e_p + jf_p$ and $Y_{pq} = G_{pq} + jB_{pq}$, and then equation (2.4) becomes $P_p - jQ_p = (e_p - jf_p) \sum_{q=1}^n (G_{pq} + jB_{pq})(e_q + jf_q)$. Separating the real and imaginary parts, we have

$$P_p = \sum_{q=1}^n [e_p(e_q G_{pq} + f_q B_{pq}) + f_p(f_q G_{pq} - e_q B_{pq})], \quad (2.5)$$

$$Q_p = \sum_{q=1}^n [f_p(e_q G_{pq} + f_q B_{pq}) - e_p(f_q G_{pq} - e_q B_{pq})]. \quad (2.6)$$

This formulation results in a set of nonlinear equations, two for each bus of the system. Note that the real and reactive powers are given by $P_p = P_{g_p} - P_{d_p}$ and $Q_p = Q_{g_p} - Q_{d_p}$, respectively, where P_{g_p} and Q_{g_p} are the generated real and reactive powers at bus p , and P_{d_p} and Q_{d_p} are the real and reactive power loads at bus p , respectively.

At this point, it is important to point out that the known real and reactive power loads P_{d_p} and Q_{d_p} present an uncertainty due the measurement errors. P_{d_p} and Q_{d_p} belong to an interval that is estimated at beginning of the process, since the accuracy of the instrument is known a priori. This implies it is necessary to admit that the real and reactive powers P_{g_p} and Q_{g_p} , which are specified in the beginning of the process, may range in an interval with an admissible radium determined by an heuristic method based on the experience of a system operator.

The real and imaginary components of voltage e_p and f_p are unknown intervals for all buses except the slack bus, where the voltage interval is specified and remains fixed. Thus there are $2(n - 1)$ equations to be solved for a load flow problem. In order to reach the solution, we use the interval version of the Newton's algorithm [6].

3. Interval Newton's Method

The problem is to find bounds on the solution of a nonlinear continuous function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ in a given box $\mathbf{x}^{(0)} \in \mathbb{I}\mathbb{R}^n$. Using the mean value theorem we have

for any x^* that $f(x^*) \in f(\tilde{x}) + J(\mathbf{x})(x^* - \tilde{x})$, where $J(\mathbf{x})$ is the interval Jacobian matrix with

$$J_{ij} = \frac{\partial^2 f}{\partial x_i \partial x_j} \quad i, j = 1, \dots, n$$

and $\tilde{x} \in \mathbf{x}$. If x^* is a zero of f then $f(x^*) = 0$ and, therefore,

$$-f(\tilde{x}) \in J(\mathbf{x})(x^* - \tilde{x}). \quad (3.1)$$

The interval linear system given by the equation (3.1) can be solved for x^* to obtain an outer bound on the solution set, say $N(\tilde{x}, \mathbf{x})$. The notation includes both \tilde{x} and \mathbf{x} to show the dependence on both terms.

It follows that $0 \in f(\tilde{x}) + J(\mathbf{x})(N(\tilde{x}, \mathbf{x}) - \tilde{x})$, which suggests the following iteration, for $k = 0, 1, \dots$ and $\tilde{x}^{(k)} \in \mathbf{x}^{(k)}$:

$$\text{solve for } N \quad f(\tilde{x}^{(k)}) + J(\mathbf{x}^{(k)})(N(\tilde{x}^{(k)}, \mathbf{x}^{(k)}) - \tilde{x}^{(k)}) = 0 \quad (3.2)$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} \cap N(\tilde{x}^{(k)}, \mathbf{x}^{(k)}). \quad (3.3)$$

A reasonable choice for $\tilde{x}^{(k)}$ is the center, denoted by \check{x} , of \mathbf{x} . In this work, we decided to use it, however other choices are available [6].

The linear system given by equation (3.2) can be solved using an appropriate interval method [6, 9] to give the Newton operator

$$N(\tilde{x}, \mathbf{x}) = \tilde{x} - J(\mathbf{x})^{-1}f(\tilde{x}).$$

We use the following notation $\Delta x = -J(\mathbf{x})^{-1}f(\tilde{x})$ that results in the iteration

$$\Delta x^{(k)} = -J(\mathbf{x}^{(k)})^{-1}f(\tilde{x}^{(k)}) \quad (3.4)$$

$$N(\tilde{x}^{(k)}, \mathbf{x}^{(k)}) = \tilde{x}^{(k)} + \Delta x^{(k)} \quad (3.5)$$

$$\mathbf{x}^{(k+1)} = \mathbf{x}^{(k)} \cap N(\tilde{x}^{(k)}, \mathbf{x}^{(k)}) \quad (3.6)$$

for $k = 0, 1, \dots$ and $\tilde{x}^{(k)} \in \mathbf{x}^{(k)}$.

4. Proposed Algorithm

In this section we briefly present the main aspects of our approach to computationally solve the power flow problem.

(i) Data Input

The data provided by the user are classified into two types: transmission-line data (resistance, reactance, susceptance of the circuits etc.) and bus data (magnitude and phase of the voltage at slack bus, generated real power at PV buses etc.). In the second group we find the load data, which are given as interval data, since they are specified considering the probable measurement errors⁵.

⁵These errors are inherent to the instruments and their tolerances are usually specified in the operation manual.

(ii) Initial Guesses

The initial guesses are provided by a punctual algorithm for the power flow problem [2], transformed into intervals that consider the maximal measurement relative error.

(iii) Applying Newton's Iteration

According to the iteration given by equations (3.4)-(3.6), we implemented the algorithm shown in Figure 1. The implementation was done in Matlab[®] using the Intlab toolbox. Notice that a real interval $x = [x_1, x_2]$ in Intlab can be stored by using either the command `infsup(x1, x2)`, where x_1 and x_2 are, respectively, the infimum and supremum of the interval x , or the command `midrad(mp, rd)`, which represents x by its midpoint $mp = \frac{x_1+x_2}{2}$ and radius $rd = \frac{x_2-x_1}{2}$. In this algorithm, `e` and `f` denote, respectively, the real and imaginary parts of the interval complex voltage at all buses, whereas `e_intv` and `f_intv` are conceived as the initial guesses, as stated in (ii). The real and reactive power are evaluated according Section 2.

The power mismatches are evaluated using equations (2.5) and (2.6) in order to obtain the box `fx`. In the given algorithm, `bsf` means a vector with all bus indices except the slack bus and `bpq` is a vector containing the bus indices for PQ buses.

After the evaluation of the Jacobian matrix⁶, the linear system given by the equation (3.4) is solved using the Matlab[®] command “\”. Then, the nonlinear system variables are updated according equations (3.5) and (3.6).

(iv) Stop Criteria

In order to assess the process convergence, we use two stop criteria. Firstly, it is verified if the box `fx` includes the zero, which means that an approximation for the solution was found. After, it is verified the possibility of improving the solution (that is, the possibility of reducing its diameter according to an admissible tolerance).

(v) Final Result in Polar Coordinates

The algorithm shown in Figure 1 provides a solution for the power flow problem in cartesian coordinates. In general, it is more usual to express the complex voltages in polar coordinates. Then, we have added a routine in order to calculate the interval polar coordinates.

5. Example System

To assess our methodology we applied to a hypothetical three-bus system composed by two load buses and a slack bus, presented in Figure 2. This network has three circuits with parameters shown in Table 1. The first and the second columns of Table

⁶In this matrix, `adj(i)` means the vector containing the bus indices of all buses connected to the i -th bus.

```

for iter = 0 : 10
    e1 = midrad(mid(e_intv), eps);
    f1 = midrad(mid(f_intv), eps);

    %----- P (real power) and Q (reactive power) -----
    E1 = e1 + j * f1;
    I1 = Y_barra * E1;
    S1 = E1 .* conj(I1);
    P1 = real(S1);
    Q1 = imag(S1);
    E = e_intv + j * f_intv;
    I = Y_barra * E;

    %----- POWER MISMATCHES -----
    dP = P1(bsf) - Pg(bsf) + Pd(bsf);
    dQ = Q1(bpq) - Qg(bpq) + Qd(bpq);

    fx = [ dP; dQ ];

    %----- STOP CRITERIA -----
    if the stop criteria are satisfied, exit

    %----- JACOBIAN MATRIX -----
    for i = 1 : nb
        K1 = e_intv(i) * G(i, i) + f_intv(i) * B(i, i);
        K2 = f_intv(i) * G(i, i) - e_intv(i) * B(i, i);
        J1(i, i) = K1 + real(I(i));
        J2(i, i) = K2 + imag(I(i));
        J3(i, i) = K2 - imag(I(i));
        J4(i, i) = - K1 + real(I(i));
        for l = adj(i) : adj(i + 1) - 1
            k = b_adj(l);
            J1(i, k) = e_intv(i) * G(i, k) + f_intv(i) * B(i, k);
            J2(i, k) = f_intv(i) * G(i, k) - e_intv(i) * B(i, k);
            J3(i, k) = J2(i, k);
            J4(i, k) = - J1(i, k);
        end
    end
    J = [ J1(bsf, bsf) J2(bsf, bsf)
          J3(bpq, bsf) J4(bpq, bsf) ];

    %----- LINEAR SYSTEM - Equation (3.9) -----
    delta = - J \ fx;

    %----- VARIABLE UPDATE -----
    e_old = e_intv;
    f_old = f_intv;
    e2 = e1(bsf) + delta(1 : nb-1); %----- Equation (3.10)
    f2 = f1(bsf) + delta(nb : 2*nb-2); %----- Equation (3.10)
    e_intv(bsf) = intersect(e_old(bsf), e2); %----- Equation (3.11)
    f_intv(bsf) = intersect(f_old(bsf), f2); %----- Equation (3.11)
end

```

Figure 1: Intlab version for the proposed algorithm.

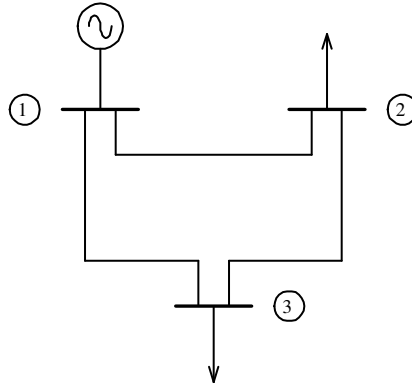


Figure 2: One-line diagram for the test-system.

2 show the bus indices between which the line is connected to. The bus data for this system are presented in Table 2, where E_{P_d} and E_{Q_d} are the relative tolerance presented by the measurement instruments that provide the real and reactive load values, respectively.

Table 1: Circuit data for the example.

From	To	Resistance (%)	Reactance (%)	Susceptance (MVar)
1	2	8.0	37.0	3.0
1	3	12.3	51.8	4.2
2	3	72.3	105.0	8.6

Table 3 shows the iteration process generated by the proposed algorithm. Iteration 0 represents the interval initial guess, which was obtained considering the solutions provided by a punctual method, as explained in Section 4.(ii). The algorithm performed 7 iterations, providing the best interval results according to the stop criteria given in Section 4.(iv), for which the tolerance adopted was 10^{-5} .

Table 4 shows the solution of the load flow study applied to the example system. According to Section 4.(v), the voltages are represented in interval polar coordinates. The mean error in the value of the magnitude V and the phase angle θ is denoted, respectively, by E_V and E_θ . Notice that the punctual values of the voltage for this system is approximately

$$V = [1.0500e+0 \angle 0.000^\circ; 9.5771e-1 \angle -9.574^\circ; 9.0373e-1 \angle -14.589^\circ]^T$$

and, therefore, we have the expected inclusions in the final interval solution presented in Table 4.

Table 2: Bus data for the example.

Bus	Type	V (pu)	Angle (°)	P_g (MW)	Q_g (MVar)	P_d (MW)	E_{P_d} (%)	Q_d (MVar)	E_{Q_d} (%)
1	2	1.05	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0	0.00	0.0	0.0	0.0	40.0	3.0	15.0	2.0
3	0	0.00	0.0	0.0	0.0	55.0	5.0	13.0	3.0

Table 3: Iterative Process.

I	e_intv	f_intv	fx
0	[1.0499e+00, 1.0501e+00] [8.9715e-01, 9.9160e-01] [8.3086e-01, 9.1833e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.2000e-02, 1.2001e-02] [-2.7480e-02, 2.7521e-02] [-2.9932e-03, 3.0069e-03] [-3.8314e-03, 3.9687e-03]
1	[1.0499e+00, 1.0501e+00] [9.2839e-01, 9.6034e-01] [8.4414e-01, 9.0497e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.1992e-02, 1.2009e-02] [-2.7506e-02, 2.7495e-02] [-3.0057e-03, 2.9944e-03] [-3.8954e-03, 3.9047e-03]
2	[1.0499e+00, 1.0501e+00] [9.3425e-01, 9.5447e-01] [8.5216e-01, 8.9695e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.1992e-02, 1.2009e-02] [-2.7506e-02, 2.7495e-02] [-3.0057e-03, 2.9944e-03] [-3.8954e-03, 3.9047e-03]
3	[1.0499e+00, 1.0501e+00] [9.3545e-01, 9.5327e-01] [8.5456e-01, 8.9455e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.1992e-02, 1.2009e-02] [-2.7506e-02, 2.7495e-02] [-3.0057e-03, 2.9944e-03] [-3.8954e-03, 3.9047e-03]
⋮	⋮	⋮	⋮
6	[1.0499e+00, 1.0501e+00] [9.3572e-01, 9.5301e-01] [8.5515e-01, 8.9395e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.1992e-02, 1.2009e-02] [-2.7506e-02, 2.7495e-02] [-3.0057e-03, 2.9944e-03] [-3.8954e-03, 3.9047e-03]
7	[1.0499e+00, 1.0501e+00] [9.3572e-01, 9.5300e-01] [8.5516e-01, 8.9395e-01]	[0.0000e+00, 0.0000e+00] [-1.6725e-01, -1.5131e-01] [-2.3902e-01, -2.1624e-01]	[-1.1992e-02, 1.2009e-02] [-2.7506e-02, 2.7495e-02] [-3.0057e-03, 2.9944e-03] [-3.8954e-03, 3.9047e-03]

Table 4: Interval Complex Voltages.

Bus	Magnitude V (pu)	E_V (%)	Phase Angle θ ($^\circ$)	E_θ (%)
1	[1.0499, 1.0501]	0,0095	[0.0000, 0.0000]	0,0000
2	[0.9410, 0.9744]	1,7438	[-10.1338, -9.0221]	5,8034
3	[0.8729, 0.9345]	3,4082	[-15.6153, -13.5988]	6,9025

6. Conclusions

This paper presented a methodology for power flow studies considering load uncertainty, using techniques of Interval Mathematics. The proposed approach was applied to several hypothetical networks (for example, the three-bus system shown in Section 5.). Punctual approaches usually presents convergence in 3 or 4 iterations. Considering the expressiveness of the information given by interval results, a convergence in 6 iterations can be considered acceptable.

The actual solution for the load flow problem with load uncertainty is indeed unknown. We can only estimate an approximation for the system voltages. In every studied cases, the final interval solution provided by our methodology, as it was expected, included the approximated punctual result of the load flow problem. The main advantage of this approach is that the analysis of the influence of the input-data errors in the final result is automatic (by the evaluation of the diameter of the interval complex voltages) e guaranteed by the definition of interval arithmetics [8].

In [13], interval techniques were also used to try to solve the power flow problem, comparing the results with those obtained by Monte Carlo simulations and by stochastic power flow. Although their methodology presented some problems, one important conclusion of that work was that interval analysis can substitute the repeated simulations required by Monte Carlo method. Some shortcomings of the stochastic approach were also pointed out, such as the problem of dealing with the non-linear relation between node loads and branch flows, and the great computational requirements that are not usually needed for conventional power flow.

Therefore, interval techniques seem to be a very good choice to deal with power flow with load uncertainty. We conclude remarking that, in actual systems, the load behavior is very difficult to be determined a priori. As further work, we intend to develop an algorithm that considers the load variation in a specified time period.

Resumo. Este artigo apresenta uma metodologia para a solução do problema do fluxo de potência em redes de energia elétrica nos quais os dados referentes à carga são incertos devido aos erros de medição nos instrumentos utilizados nas subestações. Para manipular estas incertezas, aplicam-se técnicas da Matemática Intervalar. O algoritmo utiliza a versão intervalar do método de Newton para a solução do sistema de equações não-lineares geradas pelo problema. A implementação foi realizada em Matlab[®] usando o toolbox Intlab. Para verificar o desempenho do método proposto, o algoritmo foi aplicado a sistemas de energia elétrica hipotéticos. No presente artigo, são apresentados os resultados obtidos para um sistema de três barras.

References

- [1] G. Alefeld and J. Herzberger, “Introduction to Interval Computations”, Academic Press, New York, 1983.
- [2] L.V. Barboza, “Power Flow”, in Class Notes, School of Engineering, Pelotas, 2002. (Available at <http://atlas.ucpel.tche.br/~lvb/cap3.PDF>, in Portuguese)
- [3] L.G. Dias and M.E. El-Hawary, OPF Incorporating Load Models Maximizing Net Revenue, *IEEE Trans. on Power Systems*, **8**, No. 1 (1993), 53-59.
- [4] A.F. Glimm and G.W. Stagg, Automatic Calculation of Load Flows, *AIEE Trans. on Power Apparatus and Systems*, **76**, pt. III (1957), 817-828.
- [5] C.A. Gross, “Power System Analysis”, John Wiley, New York, 1986.
- [6] G.I. Hargreaves, “Interval Analysis in MATLAB”, Numerical Analysis Report no. 416, Manchester Centre for Computational Mathematics, 2002.
- [7] R.B. Kearfott and V.Kreinovich (eds.), “Applications of Interval Computations”, Kluwer, Boston, 1996.
- [8] R.E. Moore, “Methods and Applications of Interval Analysis”, SIAM, Philadelphia, 1979.
- [9] A. Neumaier, “Interval Methods for Systems of Equations”, *Encyclopedia of Mathematics and its Applications* 37, Cambridge University Press, 1990.
- [10] S.M. Rump, IntLab - Interval Laboratory, in “Developments in Reliable Computing” (T. Csendes, ed.), pp. 77-104, Kluwer, Boston, 1999.
- [11] G.W. Stagg and A.H. El-Abiad, “Computer Methods in Power System Analysis”, McGraw-Hill Kogakusha, Tokio, 1968.
- [12] J.E. Van Ness and J.H. Griffin, Elimination Methods for Load Flow Studies, *AIEE Trans. on Power Apparatus and Systems*, **80**, pt. III (1961), 229-304.
- [13] Z. Wang and F.L. Alvarado, Interval Arithmetic in Power Flow Analysis, *Trans. on Power Systems*, **7**, No. 3 (1992), 1341-1349.
- [14] J.B. Ward and H.W. Hale, Digital Computer Solution of Power-Flow Problems, *AIEE Trans. on Power Apparatus and Systems*, **75**, pt. III (1956), 398-404.