Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

# A Normative Self-Organizing Migrating Algorithm for Power Economic Dispatch of Thermal Generators with Valve-Point Effects and Multiple Fuels

Viviana Cocco Mariani
Department of Mechanical Engineering
PPGEM, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
viviana.mariani@pucpr.br

Rodrigo Clemente Thom de Souza
Industrial and Systems Engineering Graduate Program
PPGEPS, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
usinadesolucoes@gmail.com

Leandro dos Santos Coelho
Industrial and Systems Engineering Graduate Program
PPGEPS, Pontifical Catholic University of Paraná
Imaculada Conceição, 1155, Zip code 80215-901
Curitiba, Paraná, Brazil
leandro.coelho@pucpr.br

***Abstract*— A new class of meta-heuristics called SOMA (Self-Organizing Migrating Algorithm) was proposed in recent literature. SOMA works on a population of potential solutions called specimen and it is based on the self-organizing behavior of groups of individuals in a "social environment". This paper proposes a modified SOMA approach to solving the economic load dispatch problem of thermal generators with the valve-point effect. To show the performance of the proposed modified SOMA algorithm based on fundamentals of normative knowledge in cultural algorithms, which was applied to test the power economic problem comprised 10 generating units with valve-point effects and multiple fuels for the load demands of 2400 MW. Simulation results show that the classical and modified SOMA algorithms are efficient and have good convergence property when compared with results of other optimization methods reported in the literature.**

*Keywords*— **optimization, self-organizing migrating algorithm, evolutionary computation, power systems, power generation.**

## I. INTRODUCTION

The Economic Dispatch Problem (EDP) is one of the important problems in a power system. The objective of the EDP of electric power generation is to schedule the committed generating unit outputs so as to meet the required load demand at minimum operating cost while satisfying all unit and system equality and inequality constraints [1].

In traditional EDPs, the cost function of each generator is approximately represented by a simple quadratic function and the valve-points effects ([2], [3]) are ignored. These traditional EDPs are solved using mathematical programming based on several deterministic optimization techniques.

However, the EDP problem with valve-point effects is represented as a nonsmooth optimization problem having complex and nonconvex features with heavy equality and inequality constraints [2]. This kind of optimization problem is hard, if not impossible, to solve using traditionally deterministic optimization algorithms.

As an alternative to the conventional mathematical approaches, modern stochastic optimization techniques based on evolutionary algorithms [2-5] have been given much attention by many researchers due to their ability to find potential solutions. Recently, a new class of stochastic optimization algorithm called SOMA (Self-Organizing Migrating Algorithm) was proposed in literature. SOMA is based on cooperative searching (migrating) the area of all possible solutions (search area). Individuals are mutually influenced during the search process, which leads to forming or canceling groups of individuals [6]-[8].

This paper proposes a modified SOMA (Self-Organizing Migrating Algorithm) based on normative knowledge concepts to solve the EDP associated with the valve-point effect and multiple fuels. The benchmark used to validate the modified SOMA was an EDP comprised 10 generating units with valve-point effects and multiple fuels for the load demands of 2400 MW [9]-[11]. Simulation results show that the classical SOMA and the modified SOMA were efficient when compared with the simulation results of other optimization methods reported in the literature

The remainder of this paper is organized as follows. Section II describes the formulation of the EDP, while section III explains the fundamentals of SOMA and modified SOMA approaches. Simulations and comparisons are provided in



978-1-4244-2794-9/09/$25.00 ©2009 IEEE

section IV. Finally, section V outlines the conclusion with a brief summary of results and future research.

## II. Fundamentals of Economic Dispatch Problem

The primary concern of an EDP is to minimize the total fuel cost at thermal power plants subjected to the operating constraints of a power system. Therefore, it can be formulated mathematically with an objective function and two constraints. The equality and inequality constraints are represented by equations (1) and (2) given by:

$$\sum_{i=1}^{n} P_i - P_L - P_D = 0 \quad (1)$$

$$P_i^{\min} \le P_i \le P_i^{\max} \quad (2)$$

In the power balance criterion, an equality constraint must be satisfied, as shown in equation (1). The generated power should be the same as the total load demand plus total line losses. The generating power of each generator should lie between maximum and minimum limits represented by equation (2), where $P_i$ is the power of generator *i* (in MW); *n* is the number of generators in the system; $P_D$ is the system's total demand (in MW); $P_L$ represents the total line losses (in MW) and $P_i^{min}$ and $P_i^{max}$ are, respectively, the output of the minimum and maximum operation of the generating unit *i* (in MW). The total fuel cost function is formulated as an optimization (minimization) process expressed as

$$min\ f = \sum_{i=1}^{n} F_i(P_i) \quad (3)$$

where $F_i$ is the total fuel cost for the generator unity *i* (in \$/h), which is defined by equation:

$$F_i(P_i) = a_i P_i^2 + b_i P_i + c_i \quad (4)$$

where $a_i$, $b_i$ and $c_i$ are cost coefficients of generator *i*.

In reality, the cost (objective) function of the EDP has nondifferentiable points according to valve-point loadings and multiple fuels. In this context, a cost function is obtained based on the ripple curve for more accurate modeling. This curve contains higher order nonlinearity and discontinuity due to the valve point effect, and should be refined by a sine function. Therefore, equation (4) can be modified as:

$$\tilde{F}_i(P_i) = F(P_i) + \left| e_i sin\left( f_i \left( P_i^{\min} - P_i \right) \right) \right| \quad \text{or} \quad (5)$$

$$\tilde{F}_i(P_i) = a_i P_i^2 + b_i P_i + c_i + \left| e_i sin\left( f_i \left( P_i^{\min} - P_i \right) \right) \right| \quad (6)$$

where $e_i$ and $f_i$ are constants of the valve point effect of generators. Hence, the total fuel cost that must be minimized, according to equation (3), is modified to:

$$min\ f = \sum_{i=1}^{n} \tilde{F}_i(P_i) \quad (7)$$

where $\tilde{F}_i$ is the cost function of generator *i* (in \$/h) defined by equation (6).

This paper uses an incorporated cost model, which combines the valve-point loadings and the fuel changes into one frame. The cost function should combine the equation (6) with a multiple fuels representation [9]. In this paper, it is adopted the following equations:

$$\tilde{F}_i^{MF}(P_i) = \begin{cases} a_{i1}P_i^2 + b_{i1}P_i + c_{i1} + \left| e_{i1} \cdot sin\left( f_{i1}\left( P_{i1}^{\min} - P_{i1} \right) \right) \right|, \\ \quad \text{for fuel 1}, P_i^{\min} \le P_i \le P_{i1}^{\max} \\ a_{i2}P_i^2 + b_{i2}P_i + c_{i2} + \left| e_{i2} \cdot sin\left( f_{i2}\left( P_{i2}^{\min} - P_{i2} \right) \right) \right|, \\ \quad \text{for fuel 2}, P_{i1}^{\min} \le P_i \le P_{i2}^{\max} \\ \vdots \\ a_{ik}P_i^2 + b_{ik}P_i + c_{ik} + \left| e_{ik} \cdot sin\left( f_{ik}\left( P_{ik}^{\min} - P_{ik} \right) \right) \right|, \\ \quad \text{for fuel } k, P_{ik-1}^{\min} \le P_i \le P_{i1}^{\max} \end{cases} \quad (8)$$

$$\min\ f = \sum_{i=1}^{n} \tilde{F}_i^{MF}(P_i) \quad (9)$$

where $\tilde{F}_i^{MF}$ is the cost function of generator *i* (in \$/h) defined by equation (8) for the multiple fuels (*MF*) case. In the case study presented here, we disregarded the transmission losses, $P_L$ (mentioned in equation (1)), i.e., in this work $P_L = 0$.

## III. Optimization Algorithm

This section describes the SOMA approaches. First, a brief overview of the SOMA is provided, and then the proposed modified SOMA is detailed.

### *A. SOMA algorithm*

SOMA is relatively new stochastic evolutionary algorithm proposed in [9]-[11]. SOMA is inspired on the social behavior of co-operating individuals, and self organization, i. e., competitive-cooperative strategies of individuals.

SOMA works on a population of candidate solutions (individuals) in loops called migration loops. In each loop, the population is evaluated, and the solution with the highest fitness becomes the leader. Apart from the leader, in one migration loop, all individuals will traverse the input space in the direction of the leader. An individual will travel a certain distance (called the *PathLength*) towards the leader in *n* steps of defined length. If the path length is chosen to be greater than one, then the individual will overshoot the leader [12].

The main parameters used in SOMA with their description follows [13].

*Size of the population* (*PopSize*) – number of individuals in the population.

*Dimension* – This parameter represents the number of parameters (decision variables of optimization problem) of individual.

*PathLength* – This parameter determines in what distance from the leader the actual individual will stop his movement. *PathLength*=1 means that actual individual will stop his movement on the position of the leader.

*Step* (*t*) – The step size defines the granularity with which the search space is sampled.

*Perturbation* (*PRT*) – *PRT* stands for perturbation. This parameter determines whether an individual will travel directly towards the Leader, or not.

*Migrations* (*ML*) – This parameter represents the number of *migration loops* or iterations (like generations in genetic algorithms). Their number is defined by the user, and defines for the algorithm the number of evolution cycles. The migrations number is a stopping criterion usually employed in SOMA strategies. However, the number of evaluations of the objective function is the stopping criterion adopted in this work.

The evolution of SOMA is performed using a set of stochastic evolutionary operators, which manipulate the individuals in the population. The operators in classical SOMA strategies are perturbation and crossover. Those operators are described in this section.

*Permutation operator*: In SOMA, the classical mutation used in evolutionary algorithms is replaced by the perturbation. The reason is because the movement of individuals across the hyperspace is perturbed, not mutated. Perturbation depends on the perturbation vector, which is generated according to the *PRT* parameter. For each individual's parameter, the algorithm generates a random number from the interval [0, 1]. Then, the following expression is used.

if $rnd_j < PRT$ then (10)

  $PRTVector_j = 1;$

else

  $PRTVector_j = 0;$

endif

where $j = 1, \ldots, D$, where $D$ is the number of decision variables. The perturbation vector is created before the individual starts to move across the hyperspace. The influence of the perturbation vector is on the movement. The randomly generated binary perturbation vector controls the allowed dimensions for an individual of population. If an element of the perturbation vector is set to zero, then the individual is not allowed to change its position in the corresponding dimension.

*Crossover operator*: SOMA does not create any new individuals as a combination of two or more parents. The "crossover" here means "the move of the individual across the hyperspace and map of the path of the movement". During this movement, each individual remembers the best found position, and at the end of the movement moves to this position. From this position, the individual continues in the next migration lap. The behavior of the individual during the migration is controlled by (11), or also can be controlled by (12). In this case,

$$r = r_0 + m \cdot t \cdot PRTVector \quad (11)$$

where $r$ is the new candidate solution, $r_0$ is the original individual, $m$ is the difference between the leader (individual with best fitness) and the start position of individual, $t \in$ [0, *byStepTo*, *PathLength*], and *PRTVector* is the control vector for the perturbation.

$$x_{i,j}^{ML+1} = x_{i,j,start}^{ML} + \left(x_{L,j}^{ML} - x_{i,j,start}^{ML}\right) \cdot t \cdot PRTVector_j \quad (12)$$

where $ML$ is the actual migration loop, $x_{i,j}^{ML+1}$ is the value of $j$th parameter for the $i$th individual, in step $t$ in next migration loop $ML$+1, $x_{i,j,start}^{ML}$ is the value of $j$th parameter (*start* position) of $i$th individual in current migration loop, and $x_{L,j}^{ML}$ is the value of *L*eader of $j$th parameter in migration loop $ML$.

SOMA is a stochastic optimization algorithm that is modelled on the social behaviour of co-operating individuals. In this context, SOMA is based on cooperative searching (migrating) the area of all possible solutions (search area). Individuals are mutually influenced during the search process, which leads to forming / canceling groups of individuals.

Processing in SOMA depends on the strategy used. There are several possible strategies, such as all to one, all to one rand, all to all, all to adaptive, and meta-evolution [12],[13]. The adopted strategy in this work is the *All to One Rand*. In this strategy, all individuals move to one randomly chosen leader.

In SOMA, the population is initialized randomly distributed using uniform distribution over the search space at the beginning of the search. Each parameter for each individual is generated in a given range of [*Lower*,*Upper*] boundaries, generates it according to

$$x_{i,j}^{0} = rnd_{i,j} \cdot \left(x_j^{Upper} - x_j^{Lower}\right) + x_j^{Lower} \quad (13)$$

where $i = 1, \ldots, PopSize$, and $rnd_{i,j}$ is a number generates with uniform distribution in range [0, 1].

In each loop, the population is evaluated, and the solution with the highest fitness becomes the leader $L$. Apart from the leader, in one migration loop, all individuals will traverse the input space in the direction of the leader. An individual will travel a certain distance (called the path length) towards the leader in $k$ steps of defined length.

### *B. Proposed modified SOMA (MSOMA) algorithm*

Cultural Algorithms (CAs) have been developed in order to model the evolution of the cultural component of an evolutionary computational system over time as it accumulates experience. As a result, CAs can provide an explicit mechanism for global knowledge and a useful framework within which to model self-adaptation in an evolutionary or swarm intelligence system.

In this work, in CSOMA design, new concepts of optimization are presented based on normative knowledge of CAs. The normative knowledge contains the intervals for decision variables (individuals) where good solutions have been found, in order to move new solutions towards those intervals. The $l_j$ and $u_j$ are the lower and upper bounds, respectively, for the *j*-th decision variable (parameter), and $L_j$ and $U_j$ are the values of the fitness function.

The number of individuals accepted, $\eta_{accepted}$, for update of the belief space is choice randomly of a part of population using the following expression:

$$\eta_{accepted} = p \cdot N \tag{14}$$

where *p*% is a parameter given by the user (in this work, 0.2 is adopted) and *N* is the population size. Summarizing, the update of the belief is based on the lower and upper bounds given by interval [$l_j$ , $u_j$] for the *j*-th decision variable using $\eta_{accepted}$ individuals.

In this case, the modification of equation (11) of classical SOMA proceeds as follows in the pseudo-code (Fig. 1) for the MSOMA using normative knowledge.

```
Notation: |m| :    absolute value of m
          r_j :    j-th dimension of candidate solution

Flag = 0;          Flag: boolean variable (0-false; 1-true)
If r_j < l_j then
   r_j = r_0,j + |m|·t·PRTVector;
     Flag = 1;
End if
If r_j > u_j then
     r_j = r_0,j - |m|·t·PRTVector;
     Flag = 1;
End if

If not Flag
     r_j = r_0,j + m·t·PRTVector;
End if
```

Fig. 1. Pseudo-code of MSOMA approach.

## IV. Simulation Results

When SOMA and MSOMA approaches are used for constrained optimization problems, it is common to handle constraints using the concept of penalty functions (which penalize unfeasible solutions). However, in this work is adopted a repair procedure based on [14] instead of penalizing infeasible solutions.

In order to eliminate stochastic discrepancy, in each case study, 50 independent runs were made for each of the optimization methods involving 50 different initial trial solutions for each optimization method.

For the validated benchmark, the parameters of the SOMA and MSOMA approaches are set as follows: *byStepTo* = 0.1, *PathLength* = 1.8, and *PRT* = 0.1.

The *PopSize* is set to 25 and the adopted stopping criterion was 1000 iterations (migrations) in all tested SOMA approaches. In other words, all SOMA approaches adopt 25,000 objective function evaluations in each run for the two case studies.

The results obtained for case study of multiple fuels are given in Table II, which shows that the MSOMA succeeded in finding the better solution than the SOMA for the EDP with load demand of 2500 MW. Furthermore, the low standard deviation value indicated a good convergence of MSOMA method in the 50 runs. From Table II, the mean of cost function found by CSOMA for the 50 runs carried out was better than the results of classical SOMA method. The best results obtained for solution vector $P_i$, $i$ = 1,...,10 with MSOMA with total minimum cost of 526.2298 \$/h for $P_D$ = 1800 MW is given in Table III.

## V. Conclusion

SOMA is one of the most recent stochastic methods developed for solving optimization problems. SOMA works on a population of potential solutions and it is based on the self-organizing behavior of individuals groups.

This paper proposed a MSOMA approach based on normative knowledge to solving the EDP of thermal generators with the valve-point effect. The classical SOMA and MSOMA approaches were validated for a benchmark problem consisting of 10 thermal generators with multiple fuels considering the valve point effect. The performance of the MSOMA is compared with results presented in literature for the mentioned benchmark. In this context, it is observed in terms of best result of cost function that the MSOMA have a slight advantage in terms of solution quality over the other solvers reported in recent literature (see Table IV). The best result is emphasized in boldface.

In the future, formulation of MSOMA including effect of diversity control mechanisms should be investigated in order to deal with multiobjective EDPs.

## Acknowledgment

This work was supported by the National Council of Scientific and Technologic Development of Brazil — CNPq — under Grants 302786/2008-2/PQ and 309646/2006-5/PQ.

## References

[1] T. A. A. Victoire and A. E. Jeyakumar, “Hybrid PSO-SQP for economic dispatch with valve-point effect,” *Electric Power Systems Research*, vol. 71, no. 1, pp. 51-59, 2004.

[2] C. L. Chiang, “Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels”, *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1690-1699, 2005.

[3] D. C. Walter and G. B. Sheblé, “Genetic algorithm solution of economic dispatch with valve point loading”, *IEEE Transactions on Power Systems*, vol. 8, no. 3, pp. 1325-1332, 1993.

[4] J. R. Gomes and O. R. Saavedra, “A Cauchy-based evolution strategy for solving the reactive power dispatch problem”, *Electrical Power and Energy Systems*, Vol. 24, no. 4, pp. 277-283, 2002.

[5] L. S. Coelho and V. C. Mariani, “Improved differential evolution algorithms for handling economic dispatch optimization with generator constraints”, *Energy Conversion and Management*, vol. 48, pp. 1631-1639, 2007.

[6] I. Zelinka, J. Lampinen, “SOMA - Self-Organizing Migrating Algorithm,” in *Proceedings of the 6th International Conference on Soft Computing* (*Mendel 2000*), Brno, Czech Republic, pp. 177-187.

[7] I. Zelinka, J. Lampinen, L. Noulle, “On the theoretical proof of convergence for a class of SOMA search algorithms,” in *Proceedings of 7th International Conference on Soft Computing* (*Mendel 2001*), Brno, Czech Republic, pp. 103-110.

[8] I. Zelinka, R. Senkerik, E. Navratil, “Investigation on evolutionary optimization of chaos control,” *Chaos, Solitons & Fractals* (accepted to future publication) *doi:10.1016/j.chaos.2007.07.045*.

[9] C. L. Chiang, “Improved genetic algorithm for power economic dispatch of units with valve-point effects and multiple fuels,” *IEEE Transactions on Power Systems*, vol. 20, no. 4, pp. 1690-1699, 2005.

[10] K. Y. Lee, A. Sode-Yome, J. H. Park, “Adaptive Hopfield neural networks for economic load dispatch,” *IEEE Transactions on Power Systems*, vol. 13, no 2, pp. 519-526, 1998.

[11] R. Balamurugan and S. Subramanian, “Self-adaptive differential evolution based on power economic dispatch of generator with valve-point effects and multiple fuel options,” *International Journal of Computer Science and Engineering*, vol. 1, no. 1, pp. 10-17, 2007.

[12] L. Nolle, I. Zelinka, A. A. Hopgood, A. Goddyear, “Comparison of a self-organizing migration algorithm with simulated annealing and differential evolution for automated waveform tuning,” *Advances in Engineering Software*, vol. 36, no. 10, pp. 645-653, 2005.

[13] I. Zellinka, J. Lampinen, “SOMA – self-organizing migrating algorithm,” in Proceedings of the 6th International Mendel Conference on Soft Computing, Brno, Czech Republic, 2000.

[14] N. Noman and H. Iba, “Differential evolution for economic load dispatch problems,” *Electric Power Systems Research*, vol. 78, no. 3, pp. 1322-1331, 2008.

[15] R. B. Kelkar and R. C. Desai, “Easy approach to generation scheduling with multiple fuel options,” *Journal of Inst Engrs (India)*, vol. 80, pp. 96-98, 1999.

[16] S. Baskar, P. Subbaraj and M.V.C. Rao, “Hybrid real coded genetic algorithm solution to economic dispatch problem,” *Computers and Electrical Engineering*, vol. 29, pp. 407-419, 2003.

[17] A. I. Selvakumar and K. Thanushkodi, “Anti-predatory particle swarm optimization: Solution to nonconvex economic dispatch problems,” *Electric Power Systems Research*, vol. 78, pp. 2-10, 2008.

[18] R. Balamurugan and S. Subramanian, “Hybrid integer coded differential evolution – dynamic programming approach for economic load dispatch with multiple fuel options,” *Energy Conversion and Management*, vol. 49, pp. 608-614, 2008.


TABLE II

CONVERGENCE RESULTS (50 RUNS) FOR THE CASE STUDY OF 10 GENERATING UNITS

| Method | Minimum Cost ($/h) | Mean Cost ($/h) | Maximum Cost ($/h) | Standard Deviation of Cost ($/h) |
|---|---|---|---|---|
| SOMA | 526.2428 | 526.2442 | 526.2463 | 0.0010 |
| MSOMA | 526.2298 | 526.2429 | 526.2442 | 0.0004 |

TABLE III

BEST RESULT (50 RUNS) OBTAINED USING SOMA AND MSOMA

| Power (MW) | Fuel using SOMA | Generation (MW) using SOMA | Fuel using MSOMA | Generation (MW) using MSOMA |
|---|---|---|---|---|
| $P_1$ | 2 | 207.2981 | 2 | 207.2970 |
| $P_2$ | 1 | 206.5129 | 1 | 206.2653 |
| $P_3$ | 1 | 265.5339 | 1 | 265.5339 |
| $P_4$ | 3 | 235.8770 | 3 | 236.0114 |
| $P_5$ | 1 | 258.3622 | 1 | 258.3498 |
| $P_6$ | 3 | 236.0114 | 3 | 236.0114 |
| $P_7$ | 1 | 268.7610 | 1 | 268.7594 |
| $P_8$ | 3 | 235.8770 | 3 | 236.0114 |
| $P_9$ | 1 | 332.0207 | 1 | 332.0207 |
| $P_{10}$ | 1 | 253.7458 | 1 | 253.7397 |

TABLE IV

COMPARISON OF RESULTS FOR FUEL COSTS PRESENTED IN THE LITERATURE

| Optimization Technique | $P_D$ = 2500 MW |
|---|---|
| Dynamic programming [15] | 526.240 |
| Real coded genetic algorithm [16] | 526.239 |
| Hybrid real coded genetic algorithm [16] | 526.238 |
| Improved genetic algorithm [9] | — |
| Adaptive Hopfield neural network [10] | 526.23 |
| Self-adaptive differential evolution [11] | 526.3232 |
| Anti-predatory particle swarm optimization [17] | — |
| Differential evolution combined with dynamic programming [18] | 526.2388 |
| SOMA (this paper) | 526.2428 |
| MSOMA (this paper) | **526.2298** |

TABLE I

DATA FOR THE CASE STUDY WITH 10 THERMAL GENERATORS

| Unit | Generation | | | | | | | Fuel type | Cost coefficients | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Min. | $P_1$ | $P_2$ | Max. | $F_1$ | $F_2$ | $F_3$ | | $a_i$ | $b_i$ | $c_i$ | $e_i$ | $f_i$ |
| 1 | 100 | 196 | | 250 | 1 | 2 | | 1 | $0.2697\cdot10^{2}$ | -0.3975 | $0.2176\cdot10^{-2}$ | $0.2697\cdot10^{-1}$ | -3.9750 |
| | | | | | | | | 2 | $0.2113\cdot10^{2}$ | -0.3059 | $0.1861\cdot10^{-2}$ | $0.2113\cdot10^{-1}$ | 3.0590 |
| 2 | 50 | 114 | 157 | 230 | 2 | 3 | 1 | 1 | $0.1184\cdot10^{3}$ | -1.2690 | $0.4194\cdot10^{-2}$ | 0.1184 | $-0.1269\cdot10^{2}$ |
| | | | | | | | | 2 | 1.8650 | $-0.3988\cdot10^{-1}$ | $0.1138\cdot10^{-2}$ | $0.1865\cdot10^{-2}$ | -0.3988 |
| | | | | | | | | 3 | $0.1365\cdot10^{2}$ | -0.1980 | $0.1620\cdot10^{-2}$ | $0.1365\cdot10^{-1}$ | -1.9800 |
| 3 | 200 | 332 | 388 | 500 | 1 | 3 | 2 | 1 | $0.3979\cdot10^{2}$ | -0.3116 | $0.1457\cdot10^{-2}$ | $0.3979\cdot10^{-1}$ | -3.1160 |
| | | | | | | | | 2 | $-0.5914\cdot10^{2}$ | 0.4864 | $0.1176\cdot10^{-4}$ | $-0.5914\cdot10^{-1}$ | 4.8640 |
| | | | | | | | | 3 | -2.8750 | $0.3389\cdot10^{-1}$ | $0.8035\cdot10^{-3}$ | $-0.2876\cdot10^{-2}$ | 0.3398 |
| 4 | 99 | 138 | 200 | 265 | 1 | 2 | 3 | 1 | 1.9830 | $-0.3114\cdot10^{-1}$ | $0.1049\cdot10^{-2}$ | $0.1983\cdot10^{-2}$ | -0.3114 |
| | | | | | | | | 2 | $0.5285\cdot10^{2}$ | -0.6348 | $0.2758\cdot10^{-2}$ | $0.5285\cdot10^{-1}$ | -6.3480 |
| | | | | | | | | 3 | $0.2668\cdot10^{3}$ | -2.3380 | $0.5935\cdot10^{-2}$ | 0.2668 | $-0.2338\cdot10^{2}$ |
| 5 | 190 | 338 | 407 | 490 | 1 | 2 | 3 | 1 | $0.1392\cdot10^{2}$ | $-0.8733\cdot10^{-1}$ | $0{,}1066\cdot10^{-2}$ | $0.1392\cdot10^{-1}$ | -0.8733 |
| | | | | | | | | 2 | $0.9976\cdot10^{2}$ | -0.5206 | $0.1597\cdot10^{-2}$ | $0.9976\cdot10^{-1}$ | -5.2060 |
| | | | | | | | | 3 | $-0.5399\cdot10^{2}$ | 0.4462 | $0.1498\cdot10^{-3}$ | $0.5399\cdot10^{-1}$ | 4.4620 |
| 6 | 85 | 138 | 200 | 265 | 2 | 1 | 3 | 1 | $0.5285\cdot10^{2}$ | -0.6348 | $0.2758\cdot10^{-2}$ | $0.5285\cdot10^{-1}$ | $-0.6348\cdot10$ |
| | | | | | | | | 2 | 1.9830 | $-0.3114\cdot10^{-1}$ | $0.1049\cdot10^{-2}$ | $0.1983\cdot10^{-2}$ | -0.3114 |
| | | | | | | | | 3 | $0.2668\cdot10^{3}$ | -2.3380 | $0.5935\cdot10^{-2}$ | 0.2668 | $-0.2338\cdot10^{2}$ |
| 7 | 200 | 331 | 391 | 500 | 1 | 2 | 3 | 1 | $0.1893\cdot10^{2}$ | -0.1325 | $0.1107\cdot10^{-2}$ | $0.1893\cdot10^{-1}$ | -1.3250 |
| | | | | | | | | 2 | $0.4377\cdot10^{2}$ | -0.2267 | $0.1165\cdot10^{-2}$ | $0.4377\cdot10^{-1}$ | -2.2670 |
| | | | | | | | | 3 | $-0.4335\cdot10^{2}$ | 0.3559 | $0.2454\cdot10^{-3}$ | $-0.4335\cdot10^{-1}$ | 3.5590 |
| 8 | 99 | 138 | 200 | 265 | 1 | 2 | 3 | 1 | 1.9830 | $-0.3114\cdot10^{-1}$ | $0.1049\cdot10^{-2}$ | $0.1983\cdot10^{-2}$ | -0.3114 |
| | | | | | | | | 2 | $0.5285\cdot10^{2}$ | -0.6348 | $0.2758\cdot10^{-2}$ | $0.5285\cdot10^{-1}$ | -6.3480 |
| | | | | | | | | 3 | $0.2668\cdot10^{3}$ | -2.3380 | $0.5935\cdot10^{-2}$ | 0.2668 | $-0.2338\cdot10^{2}$ |
| 9 | 130 | 213 | 370 | 440 | 3 | 1 | 3 | 1 | $0.8853\cdot10^{2}$ | -0.5675 | $0.1554\cdot10^{-2}$ | $0.8853\cdot10^{-1}$ | -5.6750 |
| | | | | | | | | 2 | $0.1530\cdot10^{2}$ | $-0.4514\cdot10^{-1}$ | $0.7033\cdot10^{-2}$ | $0.1423\cdot10^{-1}$ | -0.1817 |
| | | | | | | | | 3 | $0.1423\cdot10^{2}$ | $-0.1817\cdot10^{-1}$ | $0.6121\cdot10^{-3}$ | $0.1423\cdot10^{-1}$ | -0.1817 |
| 10 | 200 | 362 | 407 | 490 | 1 | 3 | 2 | 1 | $0.1397\cdot10^{2}$ | $-0.9938\cdot10^{-1}$ | $0.1102\cdot10^{-2}$ | $0.1397\cdot10^{-1}$ | -0.9938 |
| | | | | | | | | 2 | $-0.6113\cdot10^{2}$ | 0.5084 | $0.0416\cdot10^{-3}$ | $-0.6113\cdot10^{-1}$ | 5.0840 |
| | | | | | | | | 3 | $0.4671\cdot10^{2}$ | -0.2024 | $0.1137\cdot10^{-2}$ | $0.4671\cdot10^{-1}$ | -2.0240 |