Capacitor placement in large-sized radial distribution networks

A. Mendes, P.M. França, C. Lyra, C. Pissarra and C. Cavellucci

Abstract: The capacitor-placement problem consists of finding specific locations to install capacitor banks in an electrical distribution network. Consequently, the losses are reduced due to the compensation of the reactive component of power flow. This problem can be formulated as a nonlinear mixed-integer optimisation model and its solution has represented a challenge for many optimisation methods in the past decades. This work proposes a new method, based on evolutionary algorithms, capable of solving large network instances that appear in real-world settings. Our evolutionary approach makes use of a memetic algorithm that employs a hierarchical organisation of the population in overlapping clusters. This structure leads to special selection and reproduction schemes, which improve the algorithm's overall performance. Computational tests were executed with two small-sized instances, usually utilised as a test set in previous works, and with two real large-sized distribution networks. Tests include a sensitivity analysis of the algorithm to the optimisation's critical parameters such as the energy cost, the maximum budget available to acquire and install the capacitors, and the amortization term of the investment.

1 Introduction

Energy is continuously dissipated in electric power systems owing to electrical resistance in transmission and distribution lines. The losses in the distribution system correspond to 70% of the total losses [1]. Capacitors are widely used in such networks to reduce reactive losses due to the inductive reactive portion of the line loading. There are other beneficial effects from the application of capacitors such as power-factor correction, power-flow control and improvement of stability. However, the extent of these benefits greatly depends on how the capacitors are located on the feeder network and also their sizes. Mathematically, the capacitor-placement problem (CPP) can be formulated as a nonlinear mixed-integer optimisation problem where the objective function consists of minimising the power losses and investment costs. The main constraints comprise load constraints at each bus and operational constraints as voltage profile and current magnitudes at each node and each feeder section during varying loading levels. The nonlinear character is only due to the losses in the objective function, while integrality must be enforced because of the binary variables necessary to model the locations, the quantities and the sizes of the capacitors to be installed. For details about the mathematical model, see [2, 3].

IEE Proceedings online no. 20059015

doi:10.1049/ip-gtd:20059015

Paper first received 6th February 2004 and in revised form 25th January 2005. Originally published online: 16th June 2005

A. Mendes is with the School of Electrical Engineering and Computer Science, University of Newcastle, 2308, NSW, Australia

P.M. França, C. Lyra and C. Pissarra are with the Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, CP 6101, 13083-853 Campinas, SP, Brazil

C. Cavellucci is with the Thoth Consultoria, R. Edgard Atra 644, 13105-666, Campinas, SP, Brazil

E-mail: franca@densis.fee.unicamp.br

It is well known that the CPP is a hard combinatorial-optimisation problem, especially because real distribution networks are commonly very large and the benefits of installing capacitors in one part of the network are propagated to other parts. Hence, effective location studies should take into consideration the whole distribution system.

In the past decades, many optimisation methods have been proposed for solving the CPP. A literature survey describing the main capacitor-allocation techniques can be found in [4]. Among various approaches, the metaheuristics play a relevant role, since exact optimisation methods are not suitable for tackling real-world instances. Focusing only on metaheuristic methods, [5] proposes a 'simulated-annealing' approach. 'Tabu search' is also a possible technique capable of dealing with large instances [6, 7]. However, the most frequently used technique is based on evolutionary approaches [8–11].

This paper presents a new approach based on a genetic algorithm (GA) for solving the PCC. Its main contributions are the use of a memetic type of GA [12, 13], where a local search phase is added to the GA, and in the use of a hierarchical organisation of the population in overlapping clusters leading to special selection and reproduction schemes. The proposal also includes some practical considerations such as yearly budget restrictions that limit the amount of investment in new capacitor acquisitions. The method is capable of performing capacitor placement studies in large distribution systems, finding near-optimal solutions in a short running time.

2 Memetic-algorithm approach

In this section, the implementation of the memetic algorithm (MA) is discussed. MAs are population-based methods that can be taken as an extension of genetic algorithms (GA). The main difference between a GA and the MA implemented in this paper is that the latter includes a local search procedure, applied on the best individual at the end of every generation. Basically, an MA makes a population of solutions to evolve through the application of recombination, mutation and natural selection operators. The fitter individuals will survive longer, thus perpetuating their genetic information. After several generations, we expect the population to be composed of high-quality individuals, which represent good solutions for the CPP. Next, a simplified local search-based MA diagram is shown (Fig. 1).

Before describing these steps, the representation of individuals utilised in the MA should be illustrated. The representation chosen for the CPP uses a chromosome divided into two separated parts. The alleles of the first part can only assume binary values, coding the candidate locations' status (the sections of the feeder). If the allele in position i equals 1, it means that a capacitor should be placed in the ith feeder section, otherwise not. The second part is composed of integer values, indexing the capacitor size. Both parts of the chromosome have n positions, where n is the number of sections of the feeder. Fig. 2, shows an example of a solution for a feeder with six sections.

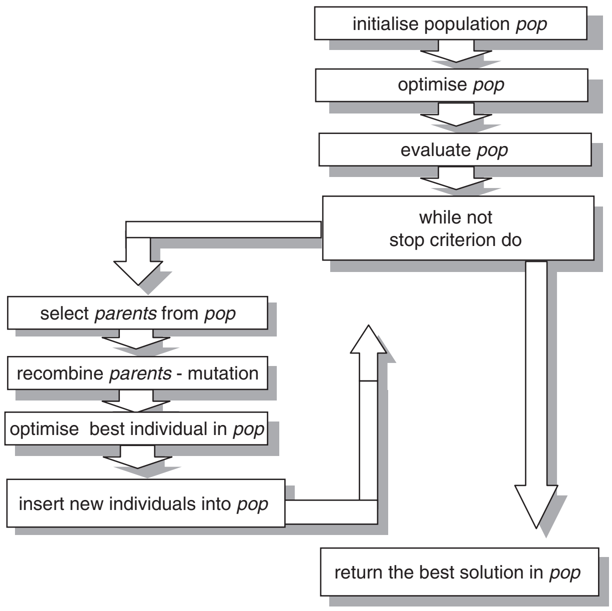

Fig. 1 *Local search-based memetic algorithm*

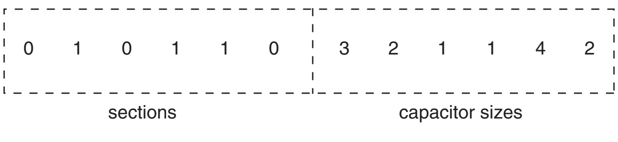

Fig. 2 *The chromosome is divided into two parts*
The first one defines whether or not a capacitor should be place at each section. The second part defines the capacitor sizes

Using the indexes defined in Table 1, Fig. 2 shows that sections 2, 4 and 5 will receive capacitors with sizes 300, 150 and 600 kVAr, respectively, and the investment cost would be US$4925. The other sections will not receive any capacitor because their correspondent alleles in the first part of the chromosome are set to zero. In such cases, the numbers in the second part of the chromosome should be ignored when calculating the investment cost and the electric power loss correspondent to this solution.

Table 1: Capacitor data

Index	Size (kVAr)	Cost (US$)
1	150	1,498
2	300	1,604
3	450	1,620
4	600	1,823
5	900	2,550
6	1200	2,955

2.1 Creating the initial population

Initially, we describe a particular feature of the population utilised in this work: the hierarchical structure. Our previous experience in solving combinatorial optimisation problems through genetic algorithms shows that the use of hierarchically structured populations leads to performance improvements over nonstructured population approaches. The hierarchy follows a ternary tree, as shown in Fig. 3. The structure can be viewed as a set of four individual clusters, each composed of a leader and three supporter individuals. In each cluster, the leader is the fittest one. This hierarchy makes the individuals at the upper levels fitter, in general, than those in the lower parts of the tree. As a consequence, the best solution is always placed at the upper cluster, at the root node of the population tree. Extensive computational tests indicated that the use of a complete three-level tree with 13 individuals was the best choice. This value might seem too low when compared with other evolutionary approaches where nonstructured populations of hundreds of individuals are usually utilised. However, the use of the hierarchy improves the evolution dynamics, allowing a major reduction in the number of individuals without loss of performance. The reduction in terms of computational complexity is overwhelming, since the algorithm will have to deal with just a few individuals, instead of literally hundreds.

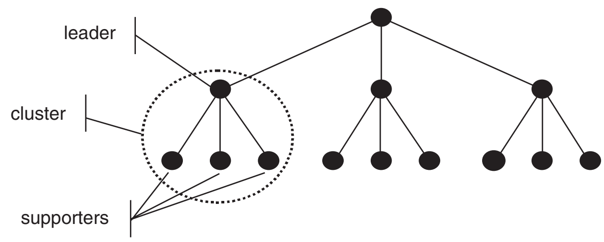

Fig. 3 *The tree-structured population is composed of several clusters, each one with four individuals*
The hierarchy relation states that every leader must be better than its supporters

The initial population is created according to the following scheme: about 20% of the sections receive a capacitor in the beginning and their sizes vary in the range [150, 1200] kVAr, with a stronger concentration between 300 and 600 kVAr. All initial individuals are randomly generated. We decided not to submit the initial individuals to an optimisation phase, as described in the MA diagram in Fig. 1. Hence, no heuristic was used to create good initial solutions by placing the capacitors in strategic locations. The MA was able to improve the quality of the randomly generated population very fast, and created good solutions in a very short time.

2.2 Selection of individuals for recombination

In the selection of individuals for recombination, first we select a leader uniformly at random. The next step is to choose which one of the three supporters will take part in the recombination. This choice is also uniformly at random. Following this selection strategy, any pair of parents will belong to the same cluster. That makes the population act similarly to a multiple-population approach with a high migration rate. Most likely, this is the cause of the algorithm's superior performance when compared to nonstructured populations. Finally, it is worth mentioning that there is no restriction to the number of times a given individual takes part in a crossover in the same generation.

2.3 Recombination

After the parents were selected following the criterion described before, they are utilised as input parameters in the recombination operator. The recombination returns a new individual—the offspring. Since the chromosome is composed of two distinct parts, they should be treated separately during the recombination process. There are, in fact, two recombination strategies: one for the chromosome's binary part and another for the integer part. In the binary part, we adopted the uniform crossover (UX), where the offspring's allele is determined by randomly choosing the value present in one of the parents. If the parents share the same allele in a given position, the offspring will inherit this value. If the values are distinct, the offspring might inherit values zero or one with the same probability. In the integer part, we calculate the average of the values found in the parents. That is, the values in the same position of the parents are added and divided by two. This will be the value inherited by the offspring. If the sum is odd, the division results in a noninteger value, which must be rounded up, or down, at random.

Figure 4 illustrates the recombination operator. In the first part of the chromosome, the offspring's positions 2 and 4 inherit the value 1, while the other positions are decided at random. In the second part, the values are calculated as averages, rounded up or down if necessary. In position 5, for instance, the parents' values are 4 and 3. The average of 3.5 was rounded up by a random decision.

Note that the most important characteristic of this operator is the maintenance of the common features of the parents. Conflicting features are resolved randomly in the binary part, and through an averaging procedure in the integer part.

The crossover rate was decided after several tests with many values, ranging from 0.5 up to 2. At the end we decided on the creation of 20 individuals per generation, corresponding to a crossover rate of 1.5. This value might seem exaggerated but, as we utilise a very strict insertion policy, many of those 20 new individuals are discarded, balancing the algorithm's dynamic. The insertion policy will be described later.

parent A	0	1	0	1	1	0	3	2	1	1	4	2
parent B	1	1	1	1	0	1	5	1	5	3	3	2
offspring	1	1	0	1	0	1	4	1	3	2	4	2

Fig. 4 *Recombination operator*

2.4 Mutation

The mutation operator aims to add diversity to the population of individuals. Similarly to the crossover, the mutation is divided into two parts. The first modifies the binary portion of the chromosome by choosing a position of the individual at random and changing the allele's value (bit-swap). The second part acts on the integer values by adding or subtracting a unity from its value. The choice of whether to add or subtract is also decided at random. Mutation is applied to 10% of the offspring. In general, higher mutation rates should be avoided because they add noise to the evolutionary process, eliminating good features already present at the chromosomes.

2.5 Local-search optimisation

After recombination and mutation, an MA submits all or some of the new individuals to a local search procedure for the purpose of improving their fitness function, as shown in Fig. 1. Many previous experiments have demonstrated the effectiveness of an MA when compared with pure GAs (i.e. without the local search phase). Our computational tests also corroborated these findings, and we are certain that the local search is a crucial step in the algorithm. Our results proved that the pure GA performs much worse than the MA, especially for large-sized problems.

In this work, we utilised three different local searches: two for the binary part for the chromosome and one for the integer part. They are applied sequentially just to the population's best individual. Next, we describe the local search policies.

2.5.1 Add/drop local search: This local search acts at the first part of the chromosome, i.e., trying to improve capacitor location. Each position of the chromosome is sequentially changed to its opposite value, and then it is verified if the fitness has improved. If a specific location already has a capacitor, the local search tests the possibility of dropping that capacitor ('drop'). Analogously, it tries to put capacitors at every location without one ('add'). In the case of deterioration of the solution, the position returns to the original value and the local search proceeds to the next one. This procedure works well when the capacitors present in the solution have not exceeded the budget limit. If the budget is already being completely utilised, no more capacitors can be added, and removing a capacitor will probably make the power loss worse.

2.5.2 Capacitors size local search: This local search acts on the second part of the chromosome. It adjusts the sizes of the capacitors already present in the solution, trying to find the best size for each location. Only the sizes immediately above and below the present capacitor's size are tested. For instance, if a 600 kVAr capacitor is installed in a given position, the procedure tries the capacitors with sizes 450 and 900 kVAr, looking for any improvement. Such tries are executed in a similar manner to the add/drop procedure, in one capacitor at a time; accepting any change that improves the fitness.

2.5.3 Swap local search: This scheme is complementary to the add/drop scheme. It acts in the first part of the chromosome, removing a capacitor from one position and installing it in another. Since it preserves the number of capacitors, it is well suited for the occasion where the budget is almost exhausted. It also acts like a fine-tuning procedure, verifying whether the already installed capacitors are placed in their best positions.

2.6 Insertion of new individuals

Each new individual created has two chances of being inserted. It may replace one of its parents—the leader of the cluster and the supporter—depending on whether or not it is better than they are. If the new individual is worse than both parents, it is discarded. This is a very strict insertion policy, which results in a high infant-mortality rate. However, together with a good mutation scheme, it allows the maintenance of diversity for longer, concurrently with a faster population's convergence rate.

Once all new individuals are created, and inserted or discarded, the algorithm starts the population-structure update. As the ternary tree must maintain the hierarchy between leaders and supporters, we verify whether any individual has become better than the leader of its cluster. In this case, they swap places. Another feature commonly adopted in MAs is elitism, which forces the presence of the incumbent solution in the population. In our implementation, this feature comes naturally from the population hierarchy, together with the policy of inserting the offspring only if it improves the population's average fitness.

2.7 Fitness function

The fitness function quantifies the quality of the individual. Therefore, it will keep a close relation with the objective function of the problem. The first factor to be observed is the cost of the power losses, which takes into account the maximum voltage deviation observed in the distribution network's nodes for a given solution. Calculation of the power losses requires the execution of a load-flow algorithm [2]. The power losses before and after the installation of capacitors are calculated and the energy gain (Δ_{gain}) is summed and then expressed as an annual gain ($Annual_{Gain}$). For this, the transformation

$$Annual_{Gain} = 8.76 * Cost_{MWh}\Delta_{gain} \tag{1}$$

is used, where $Cost_{MWh}$ is the cost, in US dollars, of the MWh at the energy market. The value 8.76 is a constant that transforms kilowatts into megawatt hours per year. It represents the number of hours in a year, divided by 1000, since Δ_{gain} is given in kilowatts and the energy cost in US dollars per megawatt hours.

Proceeding with the calculus of the fitness of the solution, the cost corresponding to the capacitor acquisitions must be subtracted from the annual gain. This cost is calculated as the sum of the costs of all capacitors to be installed. This value is also transformed into an equivalent annual cost $Annual_{Cost}$. For this it is necessary to define the amortisation term k for the equipment and an annual interest rate i. The recovery period usually corresponds to the life for computing the equipment depreciation. To do so, we utilise the classical engineering-economy transformation that, given the initial investment of the equipment Cap_{Cost}, calculates the annual payment necessary to recover the capital cost in k years with an interest rate i. Therefore the annualised cost is

$$Annual_{Cost} = (iCap_{Cost})/[1 - \{1/(1+i)^k\}] \tag{2}$$

We also included an option to restrict the annual budget available for investment in the capacitor installation, since budget constraints are usual in real situations. This constraint is controlled in the fitness function by the equation

$$Dev_A = (\max[0, Annual_{Cost} - Budget])^2 \tag{3}$$

where $Budget$ is the maximum annual budget available to install the capacitors. The square penalisation had an excellent behaviour, creating feasible individuals in the MA's first generations.

Finally, we also introduced an option to limit the number of installed capacitors. This constraint comes from operational restrictions related to the maintenance team. The equation to control the maximum number of capacitors is

$$Dev_B = (\max[0, Number_{Cap} - MaxNumber_{Cap}])^2 \tag{4}$$

where $MaxNumber_{Cap}$ is the maximum number of capacitors to be installed in the distribution network and $Number_{Cap}$ is the number of capacitors present in the solution. Finally, the fitness of the individual is then calculated as

$$fitness = Annual_{Gain} - Annual_{Cost} - Dev_A - Dev_B \tag{5}$$

2.8 Special characteristics of the implementation

The inclusion of the budget limit to be spent in capacitor acquisitions created a series of undesired disturbances caused by the local search's dynamics. An explanation for this local search behaviour follows. Suppose the method is attempting to improve an individual by using the add/drop heuristic. Since the capacitors are allocated feeder by feeder, the order is very important in this case. Suppose that the first feeder to be optimised is small and already well balanced, with a low total reactive power. The heuristic procedure will start by putting capacitors in this feeder, trying to reduce even further its reactive power. If the budget is low, there is the risk of using most of it in this initial feeder. The larger feeders, with larger power factors and that could yield better results, eventually receive fewer capacitors than they should.

This effect was partially reduced by first ordering the feeders according to their reactive/active-power ratio. This allows the optimisation to be carried out from the most reactive to the less reactive feeder, increasing the total loss reduction. Although the improvement was considerable, we noticed that the problem was to spend to the limit of the budget, no matter the feeder being optimised. The solution was to divide the budget into several parts and use it one part at a time. The strategy adopted can be described as: divide the budget into equal parts—in our case the number of parts is twice the number of feeders. The budget is thus liberated part by part, always to the feeder with the worst reactive/active-power ratio at the moment.

The next problem showed up when we examined the allocation dynamics. The sequence of positions testing was following the real network structure. It created an undesired behaviour: since the positions were indexed starting at the substations, the result was a very high concentration of capacitors close to these substations. Although this behaviour is not wrong from the theoretical point of view, it created an overall performance deterioration simply because it was too exaggerated. The solution was to examine the positions in a random sequence, which was changed every time we applied a local search. The result was a better balanced distribution of the capacitors along the network.

3 Computational tests

This section presents some numerical results obtained from computational experiments performed with the proposed method. Initially, we applied the algorithm to two small distribution networks presented in [7]. Some characteristics of these two networks are shown in Table 2. In that work, the authors use a Tabu Search technique to find the best configuration of location/size for the capacitors. The results are presented in Table 3. The column *Cost* represents the yearly power-loss cost plus the capacitor costs (in US$) and column *Time* stands for the CPU time to run the instances. This test was necessary in order to compare the MA with the best previous method available in the literature. The objective function utilised for these two instances is the same as is used in [7], which is slightly different from that presented in Section 2.7. The objective function in [7] also considers a factor that penalises the peak-load level. The results in Table 2 indicate that the MA improved the previously available results for both problems. Although the cost difference is minimal, an intriguing characteristic was unveiled. The MA tended to place the largest capacitors closer to the substation, while the TS approach did the opposite. Our result is in accord with the traditional assumption that capacitors should be always placed close to the substations. With regard to CPU time, the MA largely outperformed the Tabu Search. Fig. 5 shows the size and location of the capacitors placed by both methods in the 9-bus instance.

Table 2: Distribution-network characteristics

Network	Number of sections	Number of feeders	Initial losses
A	2,274	3	699.2 kW
B	6,865	6	901.8 kW

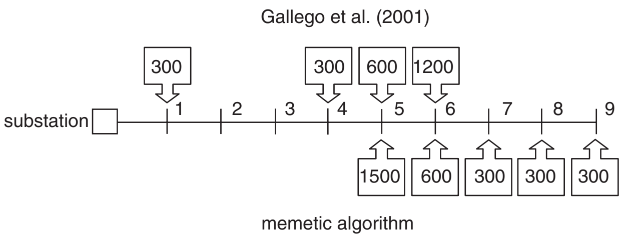

Fig. 5 *Final solutions provided by Gallego et al. (2000) and the MA for the 9-bus instance*

The next logic step was to increase dimension of the instances. To verify the potential of the method in dealing with large real-world networks, we have chosen two medium-sized Brazilian cities as case studies. The network data are shown in Table 3.

Networks *A* and *B* represent two cities with 200 000 and 500 000 inhabitants, respectively, and both are located in the state of São Paulo, Brazil. In both cities, the sections are separated by feeders: three for city *A* and six for city *B*. This separation allows the algorithm to deal with one feeder at a time, independently, thus reducing the overall computational complexity, especially in the local-search phase. To complement the study, we conducted a sensibility analysis, to test how the method would adapt itself to different scenarios of energy prices, budget limitations and amortisation terms. For each parameter, we tested four configurations, maintaining the other two at fixed values. The results are presented in Tables 4–9 in the following form:

Table 3: Results for small distribution networks

Instance	Gallego *et al.* (2001)		Memetic algorithm	
	Cost (US$)	Time (s)	Cost (US$)	Time (s)
9-bus	308,909	60	307,158	2
135-bus	192,339	300	190,446	5

Table 4: Network A: energy-price-sensitivity analysis

Energy price	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US $)
21	5850 (11)	626.5	7712
41	11 250 (33)	595.0	23 274
83	17 850 (41)	579.6	68 201
125	20 250 (51)	576.1	111 109

Table 5: Network B: energy-price-sensitivity analysis

Energy price	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US $)
21	10 950 (18)	834.3	2973
41	22 200 (39)	794.9	19 250
83	31 350 (56)	780.4	60 374
125	33 000 (75)	779.5	99 064

Table 6: Network A: annual-budget-sensitivity analysis

Annual budget	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US$)
2,083	1500 (4)	667.1	9821
4,166	3600 (9)	638.6	17 942
8,333	6600 (18)	618.2	21 465
12,500	8100 (22)	608.9	23 065

Table 7: Network B: annual-budget-sensitivity analysis

Annual budget	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US$)
2,083	2100 (4)	881.7	5288
4,166	4650 (8)	866.5	8739
8,333	8400 (17)	846.4	11 918
12,500	13 950 (24)	820.5	17 158

Table 8: Network A: amortisation-term-sensitivity analysis

Amortisation terms	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US$)
1	3300 (7)	640.8	7926
2	7950 (18)	608.1	15 618
5	11 250 (33)	595.0	23 274
10	16 500 (35)	582.1	31 856

Table 9: Network B: amortisation-term-sensitivity analysis

Amortisation term	Total capacity (no. of capacitors)	Losses (kW)	Annual profit (US$)
1	0 (0)	901.8	0
2	9750 (12)	837.7	7248
5	22 200 (39)	794.9	19 250
10	28 800 (49)	784.0	26 845

(*a*) *Total capacity:* sum of the sizes of the capacitors installed; the number of capacitors is shown in parentheses;

(*b*) *Losses in kW:* corresponds to the technical losses after the installation of the capacitors. This value should be compared with the initial power losses in Table 3, which represent the losses without the use of any capacitors;

(*c*) *Annual profit:* the total annual gain resulting from the reduction of losses after the capacitors were installed, minus the investment costs (in US dollars).

3.1 Energy-price-sensitivity analysis

The aim is to observe the behaviour of the annual profit when the price of energy varies. The parameters were set as follows: energy price (US$/MWh) from US$21 up to US$125; unlimited maximum number of capacitors; unlimited annual budget; capital recovery period of five years with an interest rate of 12% per year. This interest might seem too high at a first glance, but Brazilian companies are usually subject to such rates when financing investments in their networks' infrastructure.

In Tables 4 and 5, note that, as the energy price increases, it becomes advantageous to install more capacitors in the network. In this case, the savings resulting from the power-loss reduction more than compensates for their cost.

When the energy price reaches US$125/MWh, the method suggests the installation of 51 and 75 capacitors. These are quite large numbers, even for medium-sized cities. The inclusion in the algorithm of a constraint limiting the number of capacitors avoids this kind of outcome. Solutions with too many capacitors may create additional difficulties, since they would increase substantially the maintenance cost, not included as a cost component in our model. This maintenance cost can also be translated as a dramatic growth in human-resource requirement, since the only way to verify whether the capacitors are working nominally or not is by sending a team to test each one of them.

3.2 Maximum-budget-sensitivity analysis

The parameters were set as follows: energy price = US$41/MWh; unlimited maximum number of capacitors; annual budget ranging between US$2083 and US$12 500; the capital recovery period was set to five years (interest rate is 12% per year).

Tables 6 and 7 show how the algorithm adapts itself to the budget variation. As the budget is increased, more capacitors are installed, improving the profit. It is worth emphasising that the budget limitation can also be used as a mean to reduce the number of capacitors installed. In network *B*, for example, the MA with unlimited budget suggests the placement of 39 capacitors when the price is US$41/MWh. When the budget is limited to US$12 500, the MA suggests the use of only 24 capacitors. Although the total capacity, in kVAr, decreases nearly 37%, the annual profit is reduced in barely 11%. Another interesting characteristic is the rate of return of the invested capital. In network *A*, the annual profit ranges from two to five times the investment. In network *B*, this ratio drops considerably. In fact, although network *B* is much larger than network *A*, it seems to be less unbalanced. The addition of new capacitors yields lower percentage ratios in terms of electric losses reduction.

3.3 Amortisation term sensitivity analysis

The parameters were set as follows: the energy price is US$41/MWh; unlimited maximum number of capacitors; unlimited annual budget; amortisation term ranges from one to ten years, with an annual interest rate of 12%. Tables 8 and 9 show the results obtained.

The investment amortisation strongly influences the capacitors' costs. Since the annualised capacitor cost is reduced when the amortisation term is increased, the algorithm suggests acquiring more equipment in this situation. The longer the amortisation term, the more capacitors are installed. The number of capacitors jumped from 7 to 35 and from none to 49 capacitors in networks *A* and *B*, respectively, when the amortisation term climbed from one to ten years. The increasing is very fast, and since capacitors have a long lifetime, usually several years, real-world simulations should consider amortisation terms of five years, at least.

The memetic algorithm was programmed in C++ and a Pentium III-800 MHz was utilised. The CPU times required to solve the 9-bus and 135-bus networks were approximately 2 and 5 seconds, respectively. Networks *A* and *B* took approximately 4 min and 30 min of CPU time, respectively, for each configuration of parameters.

4 Conclusion

This paper proposes a new methodology based on genetic algorithms to deal with the capacitor-placement problem in radial-distribution electric networks. Our evolutionary approach makes use of a memetic algorithm that employs a hierarchical organisation of the population in overlapping clusters leading to special selection and reproduction schemes. Moreover, there is an intensive use of local search operators, which helped to improve the algorithm's performance.

Computational tests involved two small-sized networks. These networks were utilised as benchmarks to check the performance of the memetic algorithm against previous approaches. The new method obtained better results in both networks within less computational time.

The next step was to scale up the example size. For that, we utilised two large real-world examples. These examples represent medium-sized Brazilian cities, with over 2000 and 5000 sections where capacitors can be placed. The tests indicated that the method is a powerful and fast tool for helping planners to find out the best places to install capacitors. We also included a sensitivity analysis for some critical planning parameters such as the energy cost, the available annual budget and the amortisation term of the investment.

The results show that the new approach is able to provide expressive annual cost savings even in the presence of different parameter settings. The method is also stable, smoothly adapting itself to the different parameters, as they are changed.

5 Acknowledgments

This work was supported by Companhia Paulista de Força e Luz S. A. (CPFL—Brazil), Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq – Brazil) and Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP—Brazil). The authors would like to nominally thank to the engineers Luis René Manhães and Francisco José Bassotelli, Companhia Paulista de Força e Luz S. A. (CPFL—Brazil), for their comments on this work.

6 References

1 Bunch, J.B., Miller, R.D., and Wheeler, J.E.: 'Distribution system integrated voltage and reactive power control', *IEEE Trans.*, 1982, **PAS-101**, pp. 284–289
2 Baran, M.E., and Wu, F.F.: 'Optimal capacitor placement on radial distribution systems', *IEEE Trans.*, 1989, **PD-4**, (1), pp. 725–734
3 Baran, M.E., and Wu, F.F.: 'Optimal sizing of capacitors placed on a radial distribution systems', *IEEE Trans.*, 1989, **PD-4**, (1), pp. 735–743
4 Ng, H.N., Salama, M.M.A., and Chikhami, A.Y.: 'Classification of capacitor allocation techniques', *IEEE Trans.*, 2000, **PD-15**, (1), pp. 387–392
5 Chiang, H.D., Wang, J.C., Cokings, O., and Shin, H.D.: 'Optimal capacitor placement in distribution systems Part 2: Solution algorithms and numerical results', *IEEE Trans.*, 1990, **PD-5**, (2), pp. 643–649
6 Huang, Y.C., Yang, H.T., and Huang, C.L.: 'Solving the capacitor placement problem in a radial distribution system using tabu search approach', *IEEE Trans.*, 1996, **PS-11**, (4), pp. 1868–1873
7 Gallego, R.A., Monticelli, A.J., and Romero, R.: 'Optimal capacitor placement in radial distribution networks', *IEEE Trans.*, 2001, **PS-16**, (4), pp. 630–637
8 Sundhararajan, S., and Pahwa, A.: 'Optimal selection of capacitors for radial distribution systems using a genetic algorithm', *IEEE Trans.*, 1994, **PS-9**, (3), pp. 1499–1507
9 Miu, K.N., Chiang, H.D., and Darling, G.: 'Capacitor placement, replacement and control in large-scale distribution systems by a GA-based two-stage algorithm', *IEEE Trans.*, 1997, **PS-12**, (3), pp. 1160–1166
10 Ghose, T., Goswami, S.K., and Basu, S.K.: 'Solving capacitor placement problems in distribution systems using genetic algorithms', *Electr. Mach., Pwr. Syst.*, 1999, **27**, pp. 429–441
11 Levitin, G., Kalyuzhny, A., Shenkman, A., and Chertkov, M.: 'Optimal capacitor allocation in distribution systems using a genetic algorithm and a fast energy loss computation technique', *IEEE Trans.*, 2000, **PD-15**, (2), pp. 623–628
12 Corne, D., Dorigo, M., and Glover, F.: 'New ideas in optimization' (McGraw-Hill, 1999)
13 Burke, E.K., and Smith, A.J.: Hybrid evolutionary techniques for the maintenance scheduling problem', *IEEE Trans.*, 2000, **PS-15**, (1), pp. 122–128