DYNAMIC PROGRAMMING FOR OPTIMAL ALLOCATION OF MAINTENANCE RESOURCES ON POWER DISTRIBUTION NETWORKS

Eduardo Tadeu Bacalhau

STATE UNIVERSITY OF CAMPINAS

IFORS - Melbourne, 2011

Summary

Labore

Laboratory of Power Networks Optimisation

- Christiano Lyra Filho - State University of Campinas - christi@densis.fee.unicamp.br;
- Celso Cavellucci - State University of Campinas - celsocv@densis.fee.unicamp.br;
- Fábio Luiz Usberti - State University of Campinas - fusberti@densis.fee.unicamp.br.

Summary

Radial Network

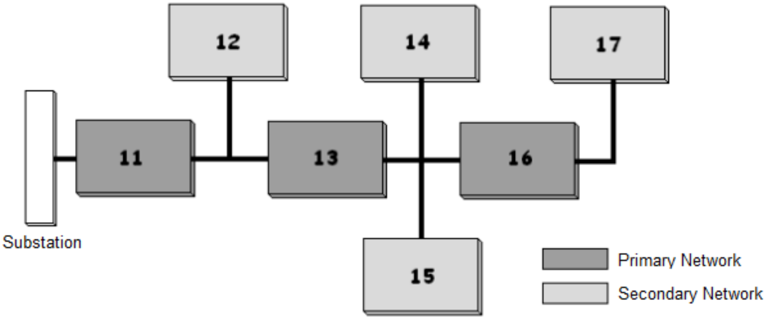


Figure: Reference Radial Network

Summary

Failure Rates

$$\lambda_s^t = \lambda_s + \sum_{e \in E_s} \lambda_e^t$$

$$\lambda_e^t = \lambda_e^{(t-1)} \sum_{n \in N_{k_e}} \delta_{k_e n} x_{en}^t$$

$$\sum_{n \in N_{k_e}} x_{en}^t = 1,$$

where:

- λ_e^t is the failure rate for equipment e in the period t;
- $\lambda_e^{(t-1)}$ is the failure rate for equipment e in the last year or the initial failure rate for equipment $e (t = 1)$;
- N_{k_e} is a set of all preventive maintenance actions;
- $\delta_{k_e n}$ is the failure rate multiplier for equipment k_e for action level n;
- x_{en}^t is a boolean decision variable denoting whether the equipment e received ($x_{en}^t = 1$) or not ($x_{en}^t = 0$) maintenance level n in the period t.

Reliability Constraints

Thus, the SAIFI (The System Average Interruption Frequency Index) of system can be calculated:

$$SAIFI^t = \frac{1}{N_T} \sum_{s \in S} \lambda_s^t N_s,$$

where:

- S is the set of all sections;
- λ_s^t is failure rate of section s in the period t;
- N_s is the number of customers into section s;
- N_T is the number of all customers into the Network.

Optimisation Problem

$$\min_{x_{en}^t} \quad \sum_{t=1}^{HP} \left\{ \sum_{e \in E} \left[\sum_{n \in N_{k_e}} (p_{k_e n} x_{en}^t) + \lambda_e^t c_{k_e} \right] \times \alpha_t \right\}$$

$$s.t: \quad SAIFI^t \leq SAIFI_{perm} \qquad \forall t = 1, \ldots HP,$$

where:

- E is a set that contains all the equipment which can receive preventive maintenance;
- $SAIFI_{perm}$ is the maximum permitted for SAIFI;
- $p_{k_e n}$ is the cost for action preventive level n for equipment k_e;
- c_{k_e} is the cost for action corrective level for equipment k_e;
- α_t is a parameter which is related to each period.

Summary

State Space Search

Example 30 equipments and HP=3

$$\mathbf{S}_0 = \begin{bmatrix} 1&0&0&1&0&1&0&1&0&1&0&1&0&1&1&0&1&1&0&1&0&1&0&1&0&1&0&1&0&0 \\ 1&0&1&1&0&0&1&0&1&0&1&0&1&0&1&0&0&0&0&0&0&0&1&1&0&0&0&0&0&0 \\ 1&1&1&0&1&0&1&0&0&0&0&0&0&0&1&1&0&0&1&0&1&0&1&0&1&0&1&0&1&1 \end{bmatrix}$$

$$Chosen\_action(\boldsymbol{S},\boldsymbol{S}')$$

- Depth Search with Simulated Annealing
  - Constructive Heuristic
  - Depth Search
  - Simulated Annealing

Summary

Knapsack Problem on Dynamic Programming

$$F_n(b) = \max \sum_{j=1}^{n} c_j x_j$$

$$s.t. \sum_{j=1}^{n} a_j x_j \leq b$$

$$x_j \in \{0, 1\}, j = 1, ..., n$$

Where:

$\sum_{j=1}^{n} c_j x_j$: total value of the selected elements to the Knapsack;

$\sum_{j=1}^{n} a_j x_j$: total volume of the selected elements to the Knapsack.

The aim is get *n-th* value as from basic cases

$$\begin{aligned} \textit{Get} \quad & F_n(a_0) \\ \textit{Where} \quad & F_k(a) = \max\{F_{k-1}(a), F_{k-1}(a - a_k) + c_k\} \\ \textit{With} \quad & F_0(a) = 0 \;\forall a \end{aligned}$$

To determine the optimal solution:

- Create an indicator p_k that is equal 0 if $F_n(b) = F_{n-1}(b)$, and 1 otherwise.
- Analyzes all indicators from p_n up to p_1. If the indicator $p_k = 0$ then $x_k^* = 0$, else $x_k^* = 1$.

Problem Adapted

$$\begin{aligned} \textit{Get} \quad & F_n(M_0) \\ \textit{Where} \quad & F_k(M) = \min\left\{F_{k-1}(M) + Cp_k + \left(\lambda_k^{cm} \times Cc_k\right),\right. \\ & \left. F_{k-1}(M - v_k) + \left(\lambda_k^{sm} \times Cc_k\right)\right\} \\ \textit{With} \quad & F_0(M) = 0 \ \forall M \end{aligned}$$

Where:

Cp_k is the maintenance preventive cost for equipment k;

Cc_k is the maintenance corrective cost for equipment k;

λ_k^{cm} is the failure rate for equipment k which was received preventive maintenance;

λ_k^{sm} is the failure rate for equipment k which was not received preventive maintenance;

v_k is the volume of the equipment k which was selected to the knapsack M.

Important Steps of the Algorithm Developed

1. Calculating the boundary given by the reliability constraints;
2. Calculating the volume of equipments:

$$v_k = \delta \times (\lambda_k^{sm} - \lambda_k^{cm})$$

3. Calculating knapsack size:

$$M = \delta \times (SAIFI_{perm} - SAIFI_{min})$$

Pseudocode

Pseudocode Knapsack Problem **(n,M)**

Calculate $SAIFI_{min}$ $SAIFI_{max}$

Calculate $v_k \ \forall\ k = 1..n$

knapsack(n,M) $\rightarrow p$

  Calculate $F_0(M) = 0 \ \forall M$

  For $k = 1..n$ **do**

    For $m = 1..M$ **do**

$$F_k(M) = \min\left\{F_{k-1}(M) + Cp_k + (\lambda_k^{cm} \times Cc_k)\ ,\ F_{k-1}(M - v_k) + (\lambda_k^{sm} \times Cc_k)\right\}$$

    End For

  End For

OptimalSolution(p,M) $\rightarrow S$

Return: S

Summary

Studied Cases

Instances

- Six intances were created for the problem:
    1. Instance with 30 equipments;
    2. Instance with 50 equipments;
    3. Instance with 100 equipments;
    4. Instance with 150 equipments;
    5. Instance with 300 equipments;
    6. Instance with 400 equipments;
- All instances were executed for only one period.

SAIFI permitted

- For each instance five values of constraints were chosen;
- Calculating via Equation:

$$SAIFI_{\alpha} = SAIFI_{min} + (SAIFI_{max} - SAIFI_{min}) \times \alpha$$

where α is $0.2,\ 0.4,\ 0.6,\ 0.8$ *and* 1.0.

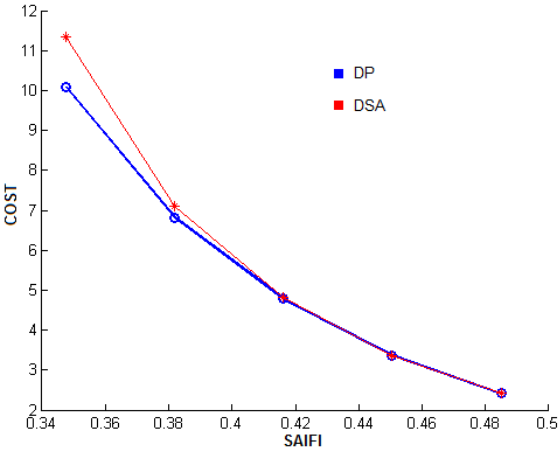
Equipments Values

Tipo	CMC	CMP	Mtx MP	CSM	Mtx SM	TF Initial
Cable	0.06	0.03	0.92	0	1.08	0.02
infrastructure 1	0.94	0.47	0.79	0	1.26	0.05
infrastructure 2	0.94	0.47	0.79	0	1.26	0.05
Post 1	14.5	7.25	0.69	0	1.2	0.001
Post 2	14.5	7.25	0.69	0	1.2	0.001
Regulator	16	8	0.89	0	1.12	0.029
Recloser	1.2	0.6	0.91	0	1.28	0.015
Primary Pruning	2.05	1.025	0.95	0	1.51	0.05
Secondary Pruning	1.05	0.525	0.95	0	1.51	0.05
Transformer	1.692	0.846	0.95	0	1.51	0.01

Results

Instance with 30 equipments:

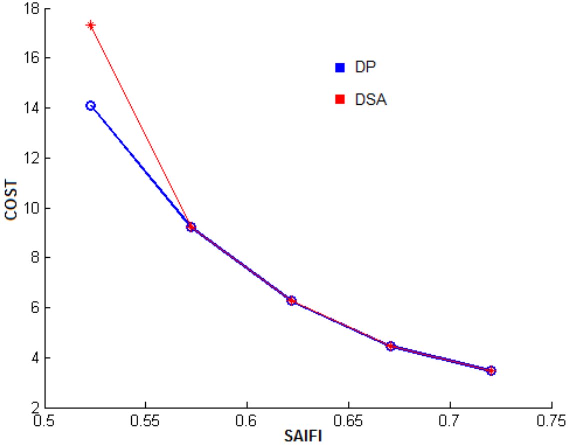
	DP		DSA		
SAIFI	Cost (x 1000)	Time (s)	Cost (x 1000)	Time (s)	Profit (%)
0.3476	**10.0766**	**0.1560**	11.3455	1.2012	11.14
0.3819	**6.8217**	0.7020	7.1146	**0.2964**	4.11
0.4163	**4.7854**	1.6224	4.8152	**0.2184**	0.61
0.4506	**3.3699**	2.8548	**3.3699**	**0.1404**	0
0.4849	**2.4145**	4.6332	**2.4145**	**0.1716**	0


12
11
10
9
8
7
6
5
4
3
2
COST
0.34
0.36
0.38
0.4
0.42
0.44
0.46
0.48
0.5
SAIFI
DP
DSA

Results

Instance with 50 equipments:

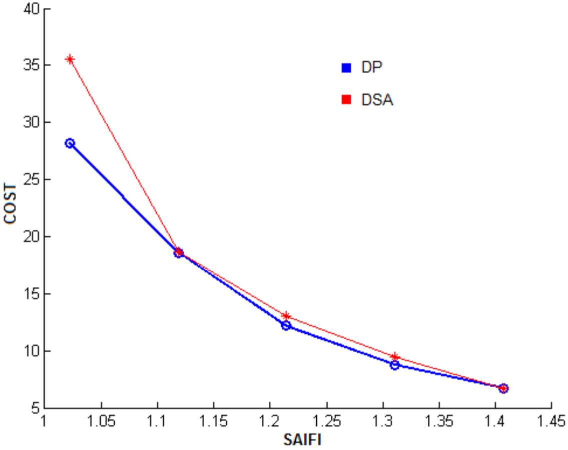
	DP		DSA		
SAIFI	Cost (x 1000)	Time (s)	Cost (x 1000)	Time (s)	Profit (%)
0.5227	**14.0922**	**0.3276**	17.3370	1.2480	18.71
0.5722	**9.2147**	1.4196	9.2326	**0.3120**	0.19
0.6216	**6.2706**	3.5256	6.3004	**0.2964**	0.47
0.6710	**4.4370**	6.3804	**4.4370**	**0.4060**	0
0.7204	**3.4518**	10.1245	**3.4518**	**0.3276**	0


DP
DSA
COST
SAIFI
18
16
14
12
10
8
6
4
2
0.5
0.55
0.6
0.65
0.7
0.75

Results

Instance with 100 equipments:

	DP		DSA		
SAIFI	Cost (x 1000)	Time (s)	Cost (x 1000)	Time (s)	Profit (%)
1.0221	**28.1844**	**0.9672**	35.5105	2.0124	20.63
1.1183	**18.5903**	4.9140	18.6201	**3.2136**	0.16
1.2144	**12.1651**	12.7141	12.9892	**1.0452**	6.34
1.3106	**8.7846**	22.1521	9.4412	**0.7020**	6.95
1.4068	**6.6941**	35.0690	6.7239	**0.9204**	0.44

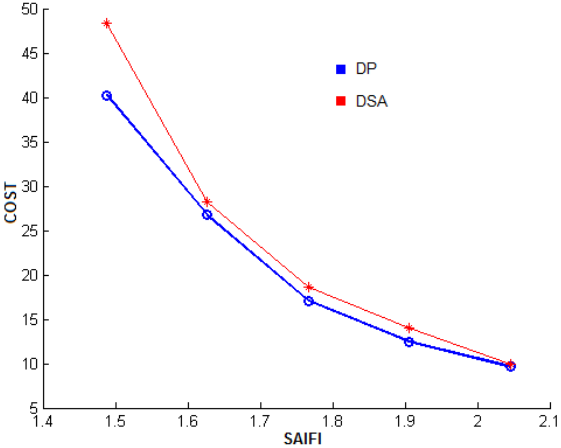


DP
DSA
COST
SAIFI
40
35
30
25
20
15
10
5
1
1.05
1.1
1.15
1.2
1.25
1.3
1.35
1.4
1.45

Results

Instance with 150 equipments:

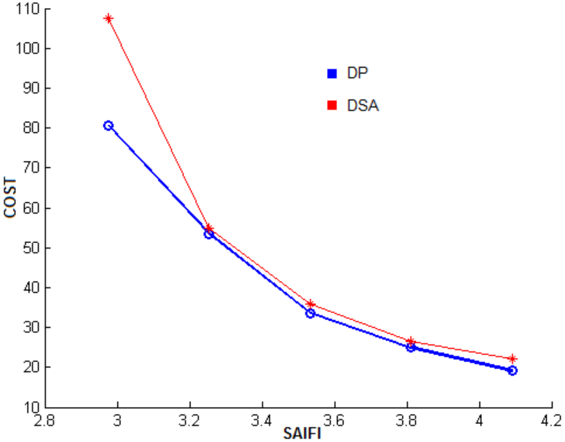
	DP		DSA		
SAIFI	**Cost** (x 1000)	**Time** (s)	**Cost** (x 1000)	**Time** (s)	**Profit** (%)
1.4878	**40.2492**	**2.1060**	48.3821	11.7157	16.80
1.6272	**26.7334**	10.7017	28.2550	**2.1216**	5.38
1.7665	**17.1013**	25.6564	18.5819	**6.6456**	7.96
1.9059	**12.4963**	47.1747	14.0188	**8.3461**	10.86
2.0452	**9.6598**	75.2237	9.9876	**13.3849**	3.28


50
45
40
35
30
25
20
15
10
5
COST
1.4
1.5
1.6
1.7
1.8
1.9
2
2.1
SAIFI
DP
DSA

Results

Instance with 300 equipments:

	DP		DSA		
SAIFI	**Cost** (x 1000)	**Time** (s)	**Cost** (x 1000)	**Time** (s)	**Profit** (%)
2.9757	**80.4985**	**11.5285**	107.6273	290.8254	25.20
3.2543	**53.4667**	**52.2447**	54.9088	250.8466	2.61
3.5330	**33.5628**	**124.7384**	35.8796	1295.7478	6.45
3.8117	**24.9627**	**242.0356**	26.6641	1681.9546	6.38
4.4068	**19.1697**	398.0834	22.2148	**231.5548**	13.70

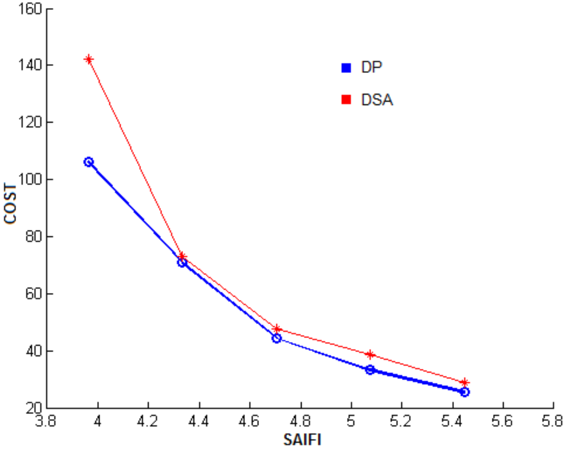


110
100
90
80
70
60
50
40
30
20
10
COST
DP
DSA
2.8
3
3.2
3.4
3.6
3.8
4
4.2
SAIFI

Results

Instance with 400 equipments:

	DP		DSA		
SAIFI	**Cost** (x 1000)	**Time** (s)	**Cost** (x 1000)	**Time** (s)	**Profit** (%)
3.9625	**106.2709**	**25.7558**	142.0483	7258.3457	25.18
4.3336	**70.8304**	**111.8215**	72.9871	705.4124	2.95
4.7046	**44.3515**	**255.7480**	47.6116	2240.4517	6.84
5.0757	**33.1114**	**446.8805**	38.4557	1290.7457	13.89
5.4468	**24.4373**	**712.9090**	28.6017	5070.5072	14.55



DP
DSA
COST
SAIFI
160
140
120
100
80
60
40
20
3.8
4
4.2
4.4
4.6
4.8
5
5.2
5.4
5.6
5.8

Summary

Conclusions

- In all instances the dynamic programming algorithm achieved the best results;
- In small instances, the state space search algorithm achieved good results when the constraints were looser, but the results deteriorates when the number of equipment grows;
- Dynamic programming has maintained a standard result on the all values of time achieved, increasing as the number of equipments grows.
- The same not happened with the state space search algorithm, increasing a lot of the computational time.

Future Works

As Future Works:

- Increase the period of optimisation;
- The development of equations dividing the problem in 2^{HP} subproblems.

Acknowledgements

- *CNPq*
- *CAPES*