

Faculdade de Engenharia da Universidade do Porto



Application of Entropy Concepts to Power System State Estimation

André Filipe Pereira dos Santos

Dissertation integrated in the
Master Degree in Electrical and Computer Engineering
Major Energy

Supervisor: Prof. Dr. Vladimiro Henrique Barrosa Pinto de Miranda

February of 2009

A Dissertação intitulada

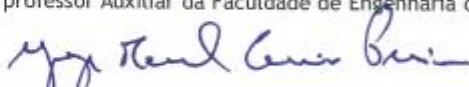
**“APPLICATION OF ENTROPY CONCEPTS TO POWER SYSTEM STATE
ESTIMATION ”**

foi aprovada em provas realizadas em 25/Fevereiro/2009

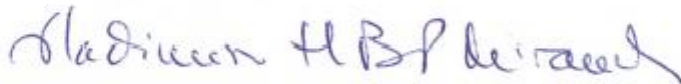
o júri



presidente Professor Doutor Carlos Manuel de Araújo Sá
professor Auxiliar da Faculdade de Engenharia da Universidade do Porto



Professor Doutor Jorge Correia Pereira
professor auxiliar da Faculdade de Economia da Universidade do Porto



Professor Doutor Vladimiro Henrique Barrosa Pinto de Miranda
professor Catedrático da Faculdade de Engenharia da Universidade do Porto

O autor declara que a presente dissertação (ou relatório de projecto) é da sua exclusiva autoria e foi escrita sem qualquer apoio externo não explicitamente autorizado. Os resultados, ideias, parágrafos, ou outros extractos tomados de ou inspirados em trabalhos de outros autores, e demais referências bibliográficas usadas, são correctamente citados

Autor - André Filipe Pereira dos Santos



Faculdade de Engenharia da Universidade do Porto

Resumo

Actualmente, existe uma necessidade crescente de explorar os sistemas eléctricos de energia (SEE) da forma mais eficiente e económica possível. Por causa do protocolo de Quioto, o aquecimento global e os subsequentes planos de redução das emissões dos gases com efeito de estufa, existe um crescimento muito grande na produção descentralizada de energia eléctrica incluindo produção eólica, solar fotovoltaica, biogás, hídrica, etc. Com a elevada quantidade de produção distribuída a ser injectada nas redes de energia existentes, existe a necessidade de operar mais próximo dos limites dos equipamentos, evitando assim, custos de investimento em novas infra-estruturas. Além disso, os SEE estão a tornar-se mais autónomos e os operadores de sistema precisam de formas mais eficientes de gerir todos os dados disponíveis dos sensores ao longo do sistema. Ademais, como os dados dos aparelhos de medida podem estar incorrectos ou até indisponíveis, é necessário utilizar algoritmos para processar essas medidas em tempo-real, de forma a obter um estado aproximado do sistema, e é aí que entra a estimação de estado em sistemas de energia. No entanto, a maioria dos modelos de estimação de estado usados actualmente, baseiam-se na minimização dos quadrados dos erros de medida - critério dos mínimos quadrados (também denominado mínimos quadrados ponderados - WLS) - como modelos de regressão. Contudo, o critério WLS apenas dá resultados óptimos se a distribuição dos erros for Gaussiana, o que não acontece na maioria dos casos, resultando em piores soluções do estimador e reduzindo, assim, a capacidade de decisão e de exploração por parte dos operadores de sistema.

Esta Tese propõe uma nova abordagem usando conceitos de entropia (conteúdo de informação dos erros), nomeadamente a correntropia, como uma melhor alternativa ao tradicional critério WLS. Com este novo conceito é possível ter presentes erros de medida grosseiros sem que estes contaminem os restantes erros, uma vez que a correntropia mostra um certo grau de insensibilidade para com os erros grosseiros. O critério da correntropia máxima (MCC) tem propriedades conhecidas de se comportar como uma métrica L0 para erros grosseiros e como uma métrica L2 para os erros pequenos (tal como o WLS). Também é testado o efeito de modulação que o tamanho das janelas de Parzen (σ) tem e na forma como este afecta a correcção dos erros e o isolamento dos erros grosseiros por parte do MCC.

Inicialmente o MCC será aplicado a uma rede DC de 4 barramentos para demonstrar como este se comporta quando comparado com os mínimos quadrados. O algoritmo de optimização utilizado é um simples método do gradiente. Seguidamente, o MCC será aplicado a uma rede AC de 24 barramentos usando um algoritmo de optimização mais robusto (EPSO) para um maior número de variáveis.

Abstract

Nowadays, there is an increasing need to explore power system networks as efficiently and economically as possible. Because of the Kyoto protocol, global warming and subsequent greenhouse gas emission reduction plans, there is a boom in decentralized power generation including wind, solar photovoltaic, biogas, hydro, etc. With the high amount of distributed generation entering the existing power system networks, there is a need to operate closer to the limits of the network equipments thus avoiding spending in new infrastructure. Moreover, power systems are becoming more autonomous and System Operators (SO) need more efficient ways to manage all data that's available from sensors throughout the system. Furthermore, since measure data can be missing or incorrect there's a need to use algorithms to process those measurements in real-time and get an approximate system state, and that's where power system state estimation comes in. However, the majority of state estimation models currently used rely on the minimization of the squared measurement errors - MSE criterion (also known as weighted least squares) as regression models. Nevertheless, the MSE criterion only gives optimum results if the errors' distribution is Gaussian and that's not the case in most problems, thus resulting in poorer results and a less effective exploration capability or decision making on the SO's part.

This Thesis proposes a new idea using entropy concepts (error information content), namely correntropy, as a better alternative to the traditional WLS criterion. With this new concept it is possible to have gross measurements present without affecting the other measurements, since correntropy shows a certain degree of insensitiveness to gross errors (outliers). The maximum correntropy criterion (MCC) has known properties of behaving as a L0 metric for large errors and L2 metric for small errors (like WLS). It is also tested the modulating effect of the Parzen window size (σ) on how it affects error correction and outlier isolation of the MCC.

Initially the MCC will be applied to a DC 4-bus toy system to demonstrate how it behaves when compared to the MSE criterion. The optimization algorithm used is the steepest ascent algorithm. Afterwards, the correntropy criterion (MCC) is applied to a 24-bus AC power system using a more robust optimization algorithm (EPSO) for large number of variables.

Keywords: Entropy, Correntropy, Evolutionary Particle Swarm Optimization, State Estimation, Minimum Square Error, Least Squares

Acknowledgements

I would like to thank Professor Vladimiro Henrique Miranda, for proposing this Thesis, for the opportunity to work with him, his endless confidence, support and continuous new ideas that enhanced this work. Furthermore, he allowed me the privilege of working alongside the skilled collaborators of the recognized institution INESC Porto (in particular, the power systems unit - *USE*).

I'd like to also thank Professor Jorge Pereira for his patience and invaluable assistance for helping me overcome some difficulties, especially with the real-time state estimation program (RTSE), developed by him.

Although not directly related to this work, I'd also like to thank Professor João Peças Lopes for his teaching methods and ideas, which inspired me and made me enjoy more and more the “energy” classes.

The development of this work was only possible due to the strength and support given to me by my family. Special thanks to my mother Ana for all the support and confidence deposited in me.

I would also like to thank all my colleagues and friends (you know who you are!), for accompanying me in this journey, and having a good time whenever was possible.

*“Nearly every man who develops an idea works it up to the point where it looks impossible,
and then he gets discouraged. That's not the place to become discouraged.”*

Thomas A. Edison

Index

Resumo	v
Abstract	vii
Acknowledgements	ix
Index	xiii
List of Figures	xv
List of Tables	xix
Abbreviations & Symbols	xxi
Chapter 1	1
Introduction	1
1.1 Problem to be addressed and its context	1
1.2 Objectives	4
1.3 Brief thesis organization	4
Chapter 2	7
State of the art	7
2.1 Introduction to power system state estimation	7
2.1.1 Overview on power systems	7
2.1.2 Analysis of power system security	8
2.1.3 Building network models	9
2.2 State Estimation formulation	13
2.2.1 DC State Estimation using WLS (MSE)	16
2.2.2 AC State Estimation using WLS (MSE)	18
2.3 Bad Data processing	21
Chapter 3	23
Entropy concepts	23
3.1 Information Theoretic Learning	23
3.2 Parzen Windows	24
3.3 Renyi's Entropy	25
3.4 Correntropy	26
3.5 State Estimation using MCC	27
Chapter 4	29
Toy DC Examples	29
4.1 Toy System	29
4.2 Example 1: Default Values	30

4.2.1 Using Parzen window size $\sigma = 0.005$	31
4.2.2 Using Parzen window size $\sigma = 1$	34
4.3 Example 2: $P_4 = -0.6$	36
4.4 Example 3: $F_{34} = -0.12$	39
4.5 Example 4: $F_{34} = 0.10$	44
4.6 Chapter conclusion.....	47
Chapter 5	49
Solving the AC problem with EPSO	49
5.1 Determining the communication probability “p”	51
5.2 Determining the Parzen window size “ σ ” and learning parameter “ τ ”.....	53
5.3 Numerical examples using AC 24-bus system	55
5.3.1 Example 1: Base case with error of 5% in line 8 (5-10).....	55
5.3.2 Example 2: Base case with error of 20% in line 8 (5-10)	61
5.3.3 Example 3: Base case with inverted measurement error in line 6 (3-9).....	67
5.4 Chapter conclusions.....	74
Chapter 6	75
Conclusions.....	75
6.1 General Conclusions	75
6.2 Future Work.....	76
References	79
Annexes.....	83
Annex A	85
IEEE Reliability Test System (AC model)	85
A.1 Network topology	85
A.2 Network parameters	86
Annex B	91
Problem data and some results	91
B.1 AC - Example 3: Line 6 gross error non-feasible solution with $\sigma=0.2$	91
Annex C	93
EPSO Algorithm	93
C.1 Genetic Algorithms.....	93
C.2 Particle Swarm Optimization (PSO)	95
C.3 Evolutionary Particle Swarm Optimization (EPSO).....	97
Annex D	99
Analysis of power system security.....	99

List of Figures

Figure 2.1 - Real-time network analysis functions [17].	9
Figure 2.2 - Functional Diagram: On-line Assessment of System Operation [14].	14
Figure 2.3 - State Estimator diagram.	16
Figure 3.1 - Three module mapper training procedure. [32]	23
Figure 3.2 - Parzen window density estimate example.	25
Figure 4.1 - Measured values of active power flows in busbars and in line 3-4 of the toy network.	29
Figure 4.2 - Absolute error comparison between MSE and MCC in default values and using $\sigma=0.005$.	32
Figure 4.3 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u.	33
Figure 4.4 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.0005$. The x-axis represents the measurement errors in p.u.	33
Figure 4.5 - Correntropy evolution curve of the steepest ascent algorithm with $\sigma=0.005$. The x-axis represents the number of iterations and the y-axis is the correntropy value being maximized.	34
Figure 4.6 - Absolute error comparison between MSE and MCC in default values and using $\sigma=1$.	35
Figure 4.7 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=1$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u.	35
Figure 4.8 - Absolute error comparison between MSE and MCC with $P_4 = -0.6$ and excluding the gross error contribution to the mean absolute error (MAE) calculation.	38
Figure 4.9 - Absolute error comparison between MSE and MCC with $P_4 = -0.6$.	38
Figure 4.10 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.007$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u.	39
Figure 4.11 - Absolute error comparison between MSE and MCC with $F_{34} = -0.12$ (inverted) and excluding the gross error contribution to the mean absolute error (MAE) calculation.	41
Figure 4.12 - Absolute error comparison between MSE and MCC with $F_{34} = -0.12$.	42

Figure 4.13 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.02$. The x-axis represents the measurement errors in p.u..	43
Figure 4.14 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..	43
Figure 4.15 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..	46
Figure 4.16 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..	46
Figure 4.17 - Absolute error comparison between MSE and MCC with $F_{34} = -0.10$ and using a new optimization parameter $\sigma=0.005$, to the mean absolute error (MAE) calculation.	47
Figure 5.1 - Network topology of the IEEE Reliability network with minor modifications.	49
Figure 5.2 - Illustration of the star communication topology (a) and the stochastic star topology (b) [46].	51
Figure 5.3 - Cooperation probability “p” (x-axis) tests with MAE values in the y-axis (lower value is better).	52
Figure 5.4 - Learning parameter “ τ ” and Parzen Window size “ σ ” tests (lower value is best). For reference, the mean absolute error (MAE) value for MSE is 0.090086. The y-axis represents the MAE.	54
Figure 5.5 - Line 8 with +5% error: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE (“Average”).	56
Figure 5.6 - Same graph as in Figure 5.5 excluding line 8 gross error. MAE recalculated.	56
Figure 5.7 - EPSO fitness correntropy function evolution of example 1. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.	57
Figure 5.8 - Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.	57
Figure 5.9 - Full Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.	58
Figure 5.10 - Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.01$. The x-axis represents error values in MW/Mvar.	58
Figure 5.11 - Full Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.	59
Figure 5.12 - Line 8 with +20% error: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE (“Average”).	61
Figure 5.13 - Same graph as in Figure 5.12 excluding line 8 gross error. MAE recalculated.	62
Figure 5.14 - EPSO fitness correntropy function evolution of example 2. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.	62
Figure 5.15 - Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.	63
Figure 5.16 - Full Gaussian error distribution of example 2 u using a Parzen window size $\sigma=0.4$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.	63

Figure 5.17 - Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.	64
Figure 5.18 - Full Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.	64
Figure 5.19 - Line 6 with error inverted: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE ("Average").	68
Figure 5.20 - Same graph as in Figure 5.19 excluding line 6 gross error. MAE recalculated..	68
Figure 5.21 - EPSO fitness correntropy function evolution of example 3. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.	69
Figure 5.22 - Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.005$. The x-axis represents error values in MW/Mvar.	69
Figure 5.23 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.005$. The x-axis represents error values in MW/Mvar.	70
Figure 5.24 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.	70
Figure 5.25 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.5$. The x-axis represents error values in MW/Mvar.	71
Figure A.1 - Network topology of the IEEE Reliability network with minor modifications. ...	85
Figure B.1 - Line 6 with inverted error: correntropy function evolution using $\sigma=0.2$	91
Figure B.2 - Line 6 with inverted error not converging: Gaussian error distribution using a Parzen window scaling parameter $\sigma'=0.2$	92
Figure B.3 - Line 6 with inverted error not converging: Gaussian error distribution using a Parzen window size $\sigma=0.2$	92
Figure C.1 - Genetic algorithm diagram scheme [49].	94
Figure C.2 - New particle position: PSO vector diagram.	96
Figure C.3 - Particle Swarm Optimization algorithm diagram.	96
Figure C.4 - Evolutionary Particle Swarm Optimization algorithm diagram.	97
Figure C.5 - New particle position: EPSO vector diagram.	98
Figure D.1 - Power system security: state diagram of system operation [14]. The gray arrows represent contingencies in the electric system.	101
Figure D.2 - Power system security: extended 6 level diagram [17; 51]. The arrows represent the involuntary level transitions caused by contingencies.	103

List of Tables

Table 4.1 - Absolute errors from MSE and MCC, in default values using $\sigma = 0.005$. Red (dark) color in <i>MCC/MSE Difference</i> means worse result for MCC when compared to MSE.	32
Table 4.2 - Absolute errors from MSE and MCC, in default values using $\sigma = 1$. Red (dark) color in <i>MCC/MSE Difference</i> means worse result for MCC when compared to MSE.	35
Table 4.3 - Absolute errors from MSE and MCC, with $P_4 = -0.6$ using $\sigma = 0.007$. Red (dark) color in <i>MCC/MSE Difference</i> means worse result for MCC when compared to MSE.	37
Table 4.4 - Comparison between MCC results in Example 1 and Example 2.	38
Table 4.5 - Absolute errors from MSE and MCC, with $F_{34} = -0.12$ (inverted) using $\sigma = 0.05$. Red (dark) color in <i>MCC/MSE Difference</i> means worse result for MCC when compared to MSE.	41
Table 4.6 - Comparison between MCC results in Example 1 ($\sigma = 0.005$), 2 ($\sigma = 0.007$) and 3 ($\sigma = 0.05$).	42
Table 4.7 - Error comparison of MCC in example 3 and MSE in example 4, with $F_{34} = 0.10$ ("exact").	45
Table 5.1 - Cooperation probability "p" tests (best value in gray). For reference, the mean absolute error (MAE) value for MSE is 0.090086.	52
Table 5.2 - Learning parameter "τ" and Parzen Window size "σ" tests (best value in gray). For reference, the mean absolute error (MAE) value for MSE is 0.090086.	53
Table 5.3 - Seed, initial measurement errors and estimated values for example 1.	60
Table 5.4 - Seed, initial measurement errors and estimated values for example 2.	65
Table 5.5 - Comparison between MCC results for example 1 and 2.	66
Table 5.6 - Seed, initial measurement errors and estimated values for example 3.	72
Table 5.7 - Comparison between MCC results for example 1 and 3.	73
Table A.1 - IEEE 24bus network line and transformer parameters ($S_b = 100\text{MVA}$).	86
Table A.2 - IEEE 24bus network load values.	87
Table A.3 - IEEE 24bus network synchronous generator parameters.	87
Table A.4 - Measurements used for the seed values, 5% and 20% error on line 5-10.	88
Table A.5 - Measurements used for the seed values and inverted error on line 3-9.	89
Table D.1 - UCTE incident: amount of cut loads by the TSO's.	100

Abbreviations & Symbols

List of abbreviations (ordered alphabetically):

CA	Contingency Analysis
DMS	Distribution Management System
EA	Evolutionary Algorithm
EMS	Energy Management System
EPSO	Evolutionary Particle Swarm Optimization
ES	Evolution Strategies
FEUP	<i>Faculdade de Engenharia da Universidade do Porto</i>
GA	Genetic Algorithms
IEEE	Institute of Electrical and Electronics Engineers
INESC	<i>Instituto de Engenharia de Sistemas e Computadores</i>
ISO	Independent System Operator
kV	kilovolt (voltage unit - $1 \text{ kV} = 10^3 \text{ V}$) [1]
MAE	Mean Absolute Error
MCC	Maximum Correntropy Criterion
MSE	Minimum Square Error (WLS)
MVA	megavoltampere (power unit - $1 \text{ MVA} = 10^6 \text{ VA}$) [1]
Mvar	megavoltampere reactive (reactive power unit - $1 \text{ Mvar} = 10^6 \text{ var}$) [1]
MW	megawatt (active power unit - $1 \text{ MW} = 10^6 \text{ W}$) [1]
OPF	Optimal Power Flow
PSO	Particle Swarm Optimization
RTSE	Real-time State Estimator (using WLS)
SCADA	System Control and Data Acquisition
SE	State Estimation
TSO	Transmission System Operator
VHV	Very high voltage (150, 220, 400 kV in Portugal)
WLS	Weighted Least Squares

List of symbols:

σ	Parzen window size (optimization parameter)
σ'	Parzen window size (graphic scaling parameter)
τ	Learning parameter (weight mutation)
h	Gradient method step
p.u.	Per Unit

Chapter 1

Introduction

This master Thesis was developed in INESC Porto, integrated in the Master Degree in Electrical and Computer Engineering at the Faculty of Engineering of the University of Porto (FEUP).

A new concept in Power System State Estimation optimization problem is presented in this work, using entropy concepts - correntropy, instead of the traditional WLS criterion. The main objective is to prove the concept and to increase awareness on this subject so that it can be further improved and developed. This approach is based on correntropy and uses, as optimization, the gradient method (DC) and the EPSO algorithm (AC).

In this chapter, the problem to be addressed will be described, including its context - State Estimation - and also the ideas that will be defended in the Thesis. Moreover, in the end of the chapter is a brief description and explanation of the thesis organization.

1.1 Problem to be addressed and its context

When the first power plants showed up in late 19th century, they were built close to industrial and populational centers and were generally connected by radial networks to the main consumption centers. Therefore, it was extremely simple to explore this kind of system by controlling turbines and alternator excitation to keep the system balanced in terms of frequency and voltage while respecting the contractual values' interval in the buses. As the industry developed and grew so did the energy demand, thus the need to increase the number of power plants, increase power in existing plants and it's respective connection to the existing grid. This increase in complexity of the power system, resulted in new control techniques being used, such as, automatic frequency control systems so that response time was faster and more adequate in the production system, thus maintaining a constant equilibrium between active power produced and consumed + losses.

For a long period of time, the essential concept in the exploration of electric power systems remained the same, i.e., controlling the production through the system frequency with some efficiency gains in individual equipment control, namely the turbine-alternator set. In addition to the frequency control, there was also the economic dispatch with the

production costs in each power plant, with system operators present in different points of the network. At that time, the power system security was essentially achieved through a high reserve margin available in the generation and transmission systems, in a way that unpredicted increases in electric energy consumption or equipment faults wouldn't cause big disturbance in the system resulting in prolonged downtimes and industrial production cuts.

This was not, by far, an ideal exploration technique thus, as the energy costs increased steeply during the last decades of the 20th century and the growth in energy consumption, with cities growing at fast rates, resulted in a significant evolution in dimension and complexity of the electric energy systems, consequently the need to find alternative exploration and control philosophies. One of the first alternatives the electric companies tried was using computer simulation to aid them in their decisions, however it soon became apparent that, since the simulation was not in real-time and the real system configuration is constantly changing, either because of faulty equipments causing changes in configuration or different load diagrams than was predicted, the simulated system configuration was very different compared to the real one. Therefore, power companies had to find a better way to explore the systems, so electric networks started being explored as a centralized system and not as small clusters or individually controlled components. This meant that the electric system would be controlled in real-time with all its inherent advantages, however they needed to mix two different control components that were previously executed separately, them being data acquisition and effective control of the system. For this idea to work they needed to send periodic information about the system variables in real-time to the control center, thus they used a system monitor to read the data from the sensors, sweeping them periodically. This allows a very good and, as most exact as possible, view of the electric system, but requires investment in good telemetry systems throughout the electric network.

Current technology is able to get good information from the network, however gross errors will certainly exist, namely due to the following:

- Bad measurement device calibration;
- Lack of measurement;
- Telemetry equipment errors due to malfunction/noise in the transmission;
- Non-simultaneous measurements;
- Measurements taken during transient incidents;
- Inaccurate network parameters;
- Network configuration errors due to errors in information (open/closed) from the switches;
- Errors in the mathematical model for not being correctly validated;
- Errors due to system asymmetries, namely unbalanced phases.

This means that the power system state observation system needs to deal with those possible bad measurements, therefore having to process all received information and compute its correct configuration and functioning system status. To solve part of this problem, some logic models were developed and are described in detail in literature [2; 3; 4]. Those are used to establish the correct system topology configuration and respective observability¹.

¹ A system is said to be observable if, for any possible sequence of state and control vectors, the current state can be determined in finite time using only the outputs.

Now that we have the correct system configuration, we need to determine the system exploration status and, to accomplish that, statistical models are implemented, in particular, using the MSE (WLS) criterion [5; 6; 7; 8] to minimize the errors' variance. For this process to work, one needs to have measurement data redundancy so that it is able to cope with the missing or bad measurements present. Moreover, this kind of process takes into account the different precisions of the measurement devices, including transducers, sensors and data communication networks.

Electric power system's reliability and security is an issue that's been increasingly more important for power companies, because of stricter regulation (through downtime/quality penalties) and also consumers that don't tolerate power outages. On the other hand, power companies, especially distribution network operators, were inefficient, thus they had to start making big investments to improve efficiency and reduce operating costs. One way to improve efficiency is with the use of real-time central control systems that allow for a better exploitation of the existing equipments.

The application of state estimation in power systems is based on the use of statistic and probability methodologies, in particular the WLS, and was initially introduced by Fred Schweppe, who published three very important articles in the late 70's [5; 6; 7]. Because of the relevance of the problem previously described, there was widespread investigation around the state estimation in power systems problem. As a result, nowadays all electric system control centers use state estimation software to cope with the power system exploration challenges.

In the 21st century there is a new paradigm in energy called *distributed generation*. Since the Kyoto protocol, its greenhouse gas emission reduction plan, and the ever increasing effects of global warming in the environment, causing extreme climate conditions, public interest in alternative energy production increased. Moreover, with the European governments' subsidies and differentiated tariffs between selling to and buying from the grid, the distributed micro-generation energy systems, such as hydro, solar photovoltaic, wind, biogas, biomass, have increasing demand. As this distributed generation is integrated in the existing power network infrastructure new challenges are presented, especially since some is very intermittent technology (wind, solar) which requires a very good communication and processing capabilities between the sensors and the control system to react to sudden changes in power generation. It is also important to forecast the evolution in demand and how much distributed power can be added to a certain point of the network without big investments.

Power system state estimation is thus facing new challenges as the network complexity and data increases and more accurate results are required. A lot has already been achieved so as to make the state estimation algorithm more robust. One of the biggest problems with state estimation is the use of MSE as estimator because, since it works by minimizing the error's variance, it is affected by the gross errors which could result in higher deviations and poorer state estimation results.

Several papers [9; 10; 11; 12; 13] have dealt with the "bad data" issue, proposing alternative solutions. This will be further detailed in the next chapter.

1.2 Objectives

In this Thesis, a new idea is proposed using entropy concepts, where state estimation using a correntropy estimator is able to “ignore” gross bad measurements and minimize the errors of the remaining measurements, in other words, it behaves as a L0 metric for large errors and a L2 metric (Euclidian norm - variance) for small errors, with the latter having similar results to the established MSE (WLS) criterion. In most of the traditionally used estimators, particularly WLS, there is a need to use additional bad measurement removal procedures that aren’t always accurate. Being able to optimize measurements in the most coherent way possible and isolate the outliers is a promising improvement over existing models.

There are also assumptions made when solving the state estimation problem with WLS, like assuming that the errors’ distribution is Gaussian. In other words, the MSE criterion can be considered an optimal approach only if the underlying distribution of the errors is Gaussian. However, that assumption isn’t verified in most cases resulting in poorer but still acceptable results.

Moreover, because of the nature of this thesis, the aim is to prove the concept and not develop a commercially efficient and robust model, especially because the optimization techniques used² are not suitable for real-time processing.

The power system state estimation with a correntropy estimator will be tested in DC and AC models, and always compared against the MSE criterion results.

1.3 Brief thesis organization

This Thesis is organized in 6 chapters and four annexes, where this introductory chapter is the first one.

In Chapter 2 a brief state of the art relating to the state estimation problem is described, including an introduction on the history of automation in power systems until the EMS/DMS modules currently used, and also a brief introduction power system security analysis. The remainder of the security analysis information, including an example of the importance of economy-security functions in the exploration of power systems illustrated with an example of a serious contingency involving the UCTE, is presented in Annex D. Additionally, some background information on network modelling construction and the characterization of the state estimation modules, including the formulation of the DC and AC models state estimation algorithms using the traditional weighted least squares regression estimator, with some observations on alternative methods for the latter. At the end of Chapter 2 a bad data analysis formulation using the Chi-Squared distribution will be described.

² Metaheuristic evolutionary strategies such as EPSO, because of their nature, are of slow convergence. Besides, the gradient method or steepest descent (or ascent) method (used to optimize neural networks’ weights) has a problem of getting stuck in local optimums and requires that the starting points are sufficiently close to the optimum solution. A solution would be to use a fast method like a Newton-Raphson, but it requires mathematical formulation and that is out of scope for this thesis.

Chapter 3 is used to introduce the Information Theoretic Learning (ITL) that is used for the manipulation of information and served as a basis for the entropy concepts. A brief review of the Parzen window, Renyi's entropy and also the correntropy functions, where the latter ones are used throughout this Thesis in Chapters 4 and 5.

In Chapter 4, a simple toy example involving a 4-bus DC network is used to test the behaviour of the state estimation using a correntropy function optimized by a gradient (steepest ascent/descent) method in four examples using different measurement errors, in order to compare those results with the MSE criterion ones. This will allow one to obtain clear results on the advantages of using the MCC over WLS, including the L0 metric for gross errors and L2 metric for the smaller ones.

In Chapter 5, a full AC model and slightly modified IEEE Reliability 24-bus test system is used. Essentially this chapter's objective is to replicate the already obtained conclusions in the previous chapter, for the DC model. However, this time an EPSO (evolutionary optimization) algorithm is used to obtain the solutions for the state estimation problem with the MCC. For the traditional WLS state estimation, an existing program was used and modified, with its estimated solutions being used as starting values for the EPSO.

Chapter 6 presents some relevant conclusions and also includes some perspectives of future work, related with the problems and new concepts introduced.

This Thesis ends with four Annexes. In Annex A, the IEEE 24-bus network data parameters and topology are presented. Annex B has some additional problem data that complement the results from Chapter 4 and 5. Annex C includes the EPSO algorithm formulation. Annex D presents a brief description of the power system security analysis.

Chapter 2

State of the art

2.1 Introduction to power system state estimation

2.1.1 Overview on power systems

“Power systems are composed of transmission, sub-transmission, distribution and generation systems” [14].

Usually, the transmission systems contain a high number of substations which are interconnected by transmission lines, transformers, and other devices for system control and protection. They are, in general, big networks in the range of dozens, hundreds or thousands of kilometers and they include a great number of sensors to keep the system well monitored.

Big power production companies are usually connected to the VHV transmission networks by its substations, and the same substations are responsible for linking the transmission network to the distribution network and other high consumers (large industries - e.g. metallurgy) that can be served directly by them. However, there is a need for voltage transformation since the generation usually has a voltage output of about 6-30kV that needs to be increased to be suitable for transmission, because the higher the voltage the more efficient is the power transmission³. In Portugal, the 3 main transmission voltages are 150, 220 and 400kV [15], and the transmission system operator (TSO) is REN - *Redes Energéticas Nacionais, SGPS*.

A characteristic of the transmission power systems is the high number of control and automation equipments. This was due to the fact that, in the past, power companies were vertically organized from generation, transmission, distribution to commercialization. Additionally, the electric utilities invested mostly in the modernization of the generation and transmission areas leading to high levels of automation and remote control in these subsystems [16]. This allows the TSO to have a real-time monitoring of the entire network through computerized systems called energy management systems (EMS).

On the other hand, the distribution systems that are usually operated in radial configuration, up until recently (last 10-20 years) had very poor investment, resulting in

³ By reducing the current and thus minimizing the copper power losses.

almost no control, nor automation. This resulted in a very low amount of sensors, thus it being impractical for the traditional real-time monitoring (like EMS) system to be used.

Therefore, the “traditional SCADA - System Control and Data Acquisition systems are being upgraded to more powerful systems that, to a certain extent, adopt some characteristics present in EMS to distribution networks”, called distribution management systems (DMS) [16].

However, in recent years there was a great evolution in how the power systems are explored. Firstly the power companies started to be divided in sub-companies; an operation called unbundling. This meant that each sub-company had to generate its own profits and become much more efficient than before, while complying with the stricter regulations on quality of service and security of demand. Besides, the new tariff schemes from the regulator have put a great pressure on the distribution power companies in order for them to increase to the investment, thus becoming more efficient and maintain an acceptable quality of service. An increase in investment translates in more automation, control and sensors in the distribution network and the more widespread use of DMS (distribution management systems). Secondly, there is now a much higher need to manage as efficiently and quickly as possible the network's information data because of the generalization of the energy markets and the subsequent need to use the obtained information for one's advantage over the competition by getting the most accurate forecasts on the energy demand, generation and possible problems that are expected. This also introduces more regulations and the gathered data must be handled and processed in the most proficient manner.

2.1.2 Analysis of power system security

Any modern society's main requirement is the economic and secure operation of its electric power system, thus regulation is getting stricter and there are increasing penalties for the failure to maintain continuity of service. To cope with this big task, there's a need to introduce and constantly evolve on the advanced large-systems analysis, optimization, and control technology.

In Annex D, a brief description of the power system security analysis is presented, just because of its importance in the exploration and management of the power systems, which includes state estimation, even though it's not directly related to the developed work in this Thesis.

2.1.3 Building network models

System operators (SO) run power systems from area control centers (e.g., the Portuguese TSO, REN, which is also the ISO⁴, is responsible for controlling the entire Portuguese network area including the interconnection with the Spanish TSO, REE (Red Electrica de España)). The system operator's primary objective is to preserve the system's normal secure state, even when contingencies occur and/or operating conditions vary throughout the day. In order to accomplish this objective, there's a need to continuously monitor the system's conditions, and consequently identify its operating state, by using the economy-security functions present in the EMS/DMS modules. Figure 2.1, illustrates the real-time economy-security analysis functions' diagram including network monitoring, contingencies analysis and also the OPF algorithms, as described by Monticelli in reference [17].

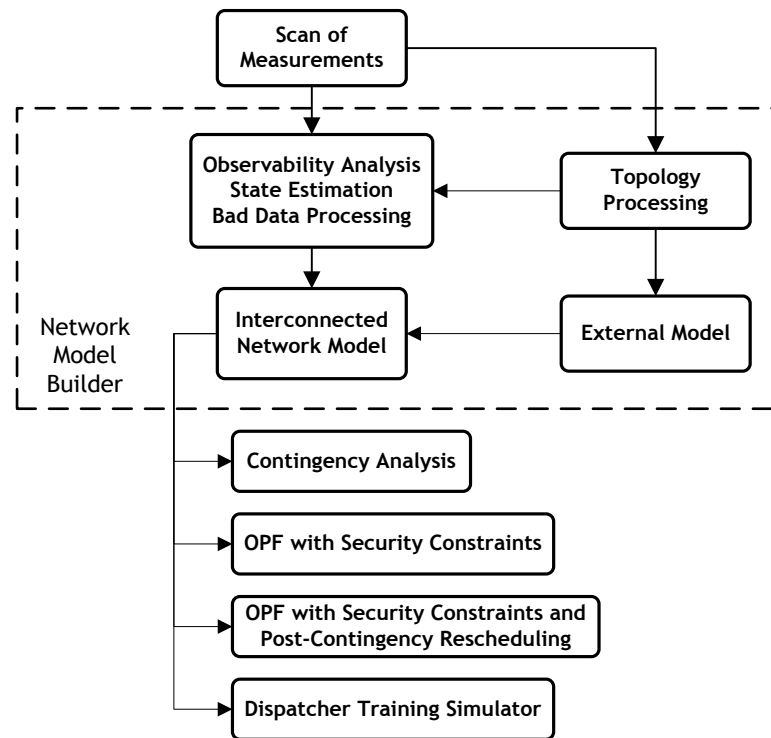


Figure 2.1 - Real-time network analysis functions [17].

The real-time processes mentioned above run on the EMS/DMS and are designed to maintain a specified security level for the lowest possible operational cost. During the exploration of the system in a pre-defined security level, it's the operator's aim to avoid entering emergency state. In case any emergency occurs, the system should be designed so that it has the capacity to recover from those emergencies.

Furthermore, simulations should be run in order to plan and adequately respond with preventive or reactive actions when contingencies arise and the system goes to an insecure state. This series of measures are usually referred to as the system's security analysis, which

⁴ Independent System Operator (ISO) is responsible for establishing, coordinating, controlling and monitoring the operation of the electrical system in its area of control.

is responsible for monitoring and determining the system's current state by processing the obtained measurements from all parts of the system.

There are two types of measurements that can be acquired by the EMS/DMS: analog and digital (equipment status - on/off (1 or 0), e.g., switches or breakers). These are then sent to the control center. The analog measurements include line power flows and current magnitudes, bus voltage magnitudes, generator outputs, loads and more recently phase and angle measurements (PMU⁵). On the other hand, digital measurements (also known as status measurements, usually with binary value), include network topology variables such as circuit breakers and switch status information, transformer tap positions, and switchable capacitor bank values. The state estimator then processes the raw measurement data with the purpose of detecting gross errors and filtering measurement noise.

Therefore, the solution obtained by the state estimator gives the best estimate of the power system state, which is directly affected by the assumed system model and available measurements. Afterwards, this information is sent to the EMS application functions, including the contingency analysis, automatic generation control (AGC), optimal power flow, etc. This information will also be made available for off-line planning and analysis functions simulations.

Power systems were originally only monitored by supervisory control systems, whose function was mainly to control and monitor the circuit breakers status on substations. The system's frequency and generation outputs were also monitored, to aid in the economic dispatch procedures and also in the controlling of the system frequency using automatic generator control mechanisms present in the generators.

However, because of the advance in computer processing power, electronic miniaturization, increasing sophistication of sensors and the need for the power companies to be more efficient, including the facilitation of the security analysis, made the supervisory control systems per-se obsolete. Thus, real-time system-wide data acquisition capabilities were added to the existing control systems, which allowed for analog and digital measurements (switches and circuit breakers status data) to be collected by the control centers.

This new system was named Supervisory Control and Data Acquisition (SCADA) and this was a needed upgrade to the existing supervision system because several of the economy-security functions like the contingency analysis, corrective dispatch in both active and reactive power needed to be executed with real-time operating conditions of the system defined and available. However, the presence of errors in the measurements, missing data, or communication interferences could make the SCADA system unreliable. Moreover, the collected set of measurements, like absence of line flows or voltage phase angles that aren't used very often may not allow a direct extraction of the corresponding AC system's operating state. Also, using all possible measurements might not even be doable on cost-effective terms.

⁵ Synchronized Phase Measurement Units (PMU) were introduced in the mid-1980s [18; 19] and are gaining more attention by researchers, and very recently in [20] a proposed a novel method for the measurement of the rotor angles and the phasors of power systems. In the traditional SCADA/EMS systems, only the magnitudes are measured, and the phases are estimated using the state estimation algorithm. However, estimated phase may not be accurate enough and in order to improve the state estimation procedure, phases should also be measured.

Hence, all these issues mean that there is a need for pre-processing the received measurements and data firstly doing a “Topology Processing” and an “Observability Analysis”, just like is shown in Figure 2.1, in order to define the system topology using switching and breaker data and, lastly, see if the obtained measurements allow for a complete system observability or not. The next step is to compute the most accurate system state using the “State Estimator” application, which will be explained in more detail in subsequent sub-chapters.

The network model can thus be defined as an up-to-date network state including the presence of the analog measurements (e.g., power flows, loads and generations) and the digital ones (network topology variables).

In reference [16], the author reports that the network model is normally obtained from scanning the measurement’s values in a periodic time sequence of every one or two seconds, which allows for the EMS/DMS to compute the model of the network in real-time.

The first step in real-time modelling is the creation of dynamic and static databases, where the first one is included inside SCADA and is responsible for processing both the analog and digital measurement data mentioned before, and the latter contains fixed information about the network parameters, like the impedances from lines, transformers, etc. and also the configuration data (e.g. connections among the equipments). After this step, network observability is tested, using both databases’ information. Therefore, if the state variables can be evaluated, the network is considered observable. Even if observable islands are created by the lack of telemetry or erroneous data, pseudo-measurements can be used to obtain a full state estimation. However, in this case, care must be taken to avoid corrupting the estimations from the telemetry data.

Figure 2.1, illustrates the needed steps for the network model construction after receiving all the scanned measurements.

Finally, the problem with the interconnected networks also needs to be addressed.

This is an issue that EMS and DMS have when they are building the network model, because if there are interconnections with other system control areas outside of their responsibility (e.g. REN - REE control areas, which connect Portugal and Spain) and there’s a deficient communication between them, some approximations may have to be used, such as, a reduced network equivalent. Also, changes in an adjacent external network area topology (e.g. line or generator tripping) could greatly affect the operator’s own network. For these reasons it’s required that a high degree of communication is always maintained between all the interconnected system operators for joint systems operation management, giving the example of the UCTE network presented in Annex D. If there was no communication between the TSO’s it wouldn’t be feasible to operate in cooperation like it’s done. This way, the system can be operated in an efficient and secure way.

The power network controlled by the EMS/DMS is usually referred as the Internal System⁶ and the adjacent interconnected networks from different ownership are called External Systems. The latter ones can be replaced by reduced equivalent networks or can be maintained in the unreduced form.

⁶ Also known as Own System (OS). Most commonly referenced, simply as “power system”.

As a result, to build the model for the interconnected systems, three main approaches are presented in reference [17]:

- i. **Power flow based method:** here the external model (reduced or unreduced) is attached to the internal model using boundary matching injections. These injections are then computed by solving a power flow for the external network while treating the internal boundary buses as slack buses. This means that the voltage values (magnitudes and angles) in these buses are obtained by executing the power system's state estimation, using them as specified values. Therefore, the obtained interconnected network model is able to reproduce the conditions of the internal system, because it is able to completely absorb the possible external errors. But, even though the internal system is correctly reproduced, care needs to be taken because the same external errors could affect the contingency analysis and also the OPF studies, since the reactions of the external model to changes (i.e. contingencies like short-circuits, etc.) in the internal one, might be erroneous. In references [21; 22; 23], the authors use this method on some test networks in order to evaluate its behaviour. Also, in [24] the author A. Bose used a combined unreduced load flow model and an external network equivalent;
- ii. **One-pass state estimation technique:** this method runs a single state estimation pass for the whole network, including both the internal and external systems. Besides, the power flow variables (for example, the specified values for voltages and flows, etc.) and its corresponding limits (for example, the reactive power generation limits, node voltage limits, etc.) are being considered as pseudo-measurements. However some caution should be used especially when dealing with low impedance lines, because a lack of measurements on their power flows or the power injections in its terminal nodes could result in large variances, thus possible contamination of the results. A possible solution suggested by the author is to assign a pseudo-measurement on every low impedance branch, which can be done without affecting the overall measurement redundancy. In references [25; 26], a single state estimation is performed for both the internal and external networks, and some numerical instability problems are detected. But, in [27] an alternative one-pass method is used, based on a set of critical external pseudo-measurements that gives better results;
- iii. **Two-pass state estimation technique:** lastly, this method uses state estimation to give an initial estimate of the internal system states. The external model is then attached in two passes. The first, uses an initial power flow, like on $i.$, in order to calculate the branch power flows in the unobservable network. The second, runs a state estimation using the estimated states for the internal system as pseudo-measurements, and the calculated power flows for the external system. The scheduled power injections at the inner boundary nodes are used as target values. Also, zero injections for both the internal and external networks are also treated as pseudo-measurements. Furthermore, if external real-time telemetry is available, it can be included in the second pass of the state estimation. In references [28; 29], both approaches are able to reduce the effects of the boundary errors in the internal system solution, by proper weighting of the external system's pseudo-measurements. Additionally, a partitioned state estimation method for real-time external network modelling is presented in [30].

2.2 State Estimation formulation

There is a conventional methodology for the state estimation which served as basis for almost all the other methodologies. This methodology was introduced in three papers [5; 6; 7] by Fred Schweppe et. al. However, the denomination of state estimation can have an extensive connotation, because it can involve, not only, the evaluation algorithm but also the entire network model builder. The introduction of the state estimation function expanded the capabilities of the SCADA system computers, which lead to the emergence of the Energy Management Systems (EMS), which were equipped with an on-line state estimator among other applications. Therefore, the most common state estimators include the following functions [14]:

- **Topology Processor:** the first step in the state estimation process and is responsible for obtaining status data (digital measurements) from circuit breakers and switches, and, with them, building a coherent one-line diagram of the system;
- **Observability Analysis:** after having the topology defined, a next step is needed to check if any unobservable branches or observable islands exist, using the previously set of scanned measurements. This is done by checking if a state estimation solution for the entire system can be obtained from the existing measurements;
- **State Estimation solution:** is responsible for calculating the best estimate for the power system state. It receives as input the analog measurements and also the assumed network model, and returns as output, the complex bus voltages in the entire power system, and also the best estimates for all the line flows, loads, transformer taps, and generation outputs. Note that this process is only effective if there is a redundancy of measurements, otherwise if the number of measurements was the same as the number of state variables the algorithm would behave as a traditional power flow;
- **Bad Data Processing:** this module's objective is to detect and eliminate the gross measurement errors. Again, this can only be done effectively if the number of measurements is greater than the number of state variables. This is usually a probabilistic method, in that it uses functions such as the Chi-Squared distribution to catch the most likely bad measurements;
- **Parameter and structural error processing:** estimates various network parameters, such as transmission line model parameters, tap changing transformer parameters, shunt capacitor or reactor parameters. Detects structural errors in the network configuration and identifies the erroneous breaker status provided that there is enough measurement redundancy.

Power system state estimator constitutes the core of the on-line security analysis function, as it is able to filter the received raw measurements and provide the most reliable data base for all the EMS/DMS applications that require them. This model includes all buses within the area of the control center EMS/DMS and chosen buses from the neighbouring systems (external systems). Besides, measurements and additional data obtained from the neighbouring systems are used to create and continuously update the external system model.

Moreover, there may be unobservable parts of the system within own operator area, caused by telemetry problems, rejected bad data or other unexpected malfunctions. As described before, some of the used data will be estimated using pseudo-measurements. Those are obtained from short term load forecasts, generation dispatch, previous records or other similar approximation methods. Because of their nature, one usually gives them low weights because of their inherent high variances, to prevent contamination of the other measurements, or they can be forced as critical measurements if desired.

Besides, passive buses with no generation nor load, thus having zero active and reactive power injection could be present. If those are used, even though they are not actually measured, one can trust them as error free measurements in the state estimation formulation. These are named "virtual" measurements. The obtained solutions from the state estimator will be evaluated, so that a system state classification is selected, as can be seen in Figures D.1 and D.2., Contingency Analysis should be run if the system is on normal state, in order to evaluate the system's security against a number of pre-defined contingencies. If the system is found to be insecure, then some preventive measures need to be calculated using software functions, like a security constrained OPF. After the implementation of the calculated preventive measures, the system will go the normal and secure state.

Figure 2.2 illustrates the whole security assessment of the real-time analysis of the power system, including the emergency and restorative control actions that should only be reached in extreme conditions, especially the latter.

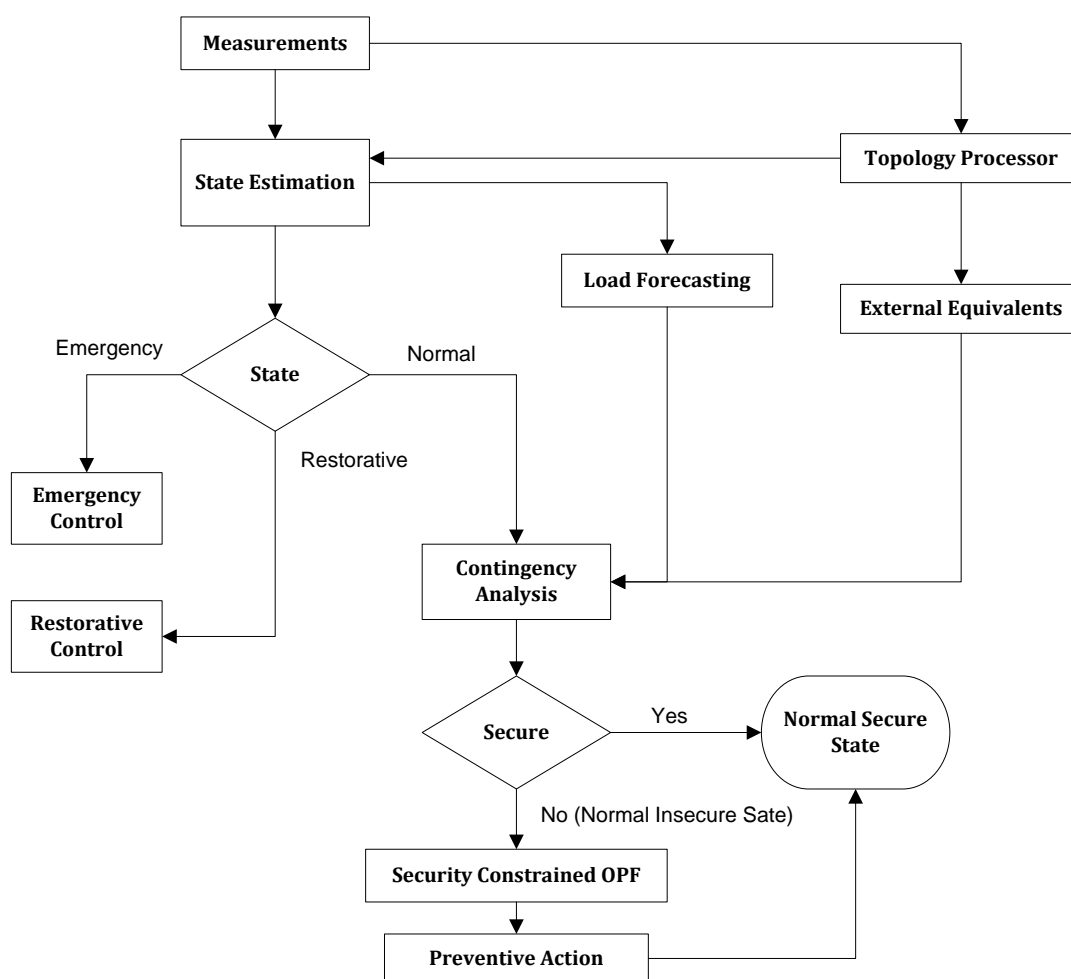


Figure 2.2 - Functional Diagram: On-line Assessment of System Operation [14].

In sum, this methodology introduces several advantages, including:

1. Supplies complete real-time information of the system's behaviour;
2. Allows better knowledge of the system's limits (constraints) continuously, e.g. if line power limit has sufficient slack;
3. Facilitates the decision making, reducing the response time and minimizing the risk of wrong decisions being made;
4. Allows for the detection of malfunctions in the system.

However, state estimators have to deal with some issues, including bad data analysis procedures that will be discussed in more detail in one of the subsequent sub-chapters. Some of these issues are:

- Existence of measurement errors, whether caused by equipment noise or malfunction, including telemetry errors;
- Errors caused by bad calibration of the devices;
- Absence of measurements or non-simultaneous readouts;
- When measurements are obtained during a transient phenomenon;
- Errors caused by asymmetries in the system or in the mathematical models;
- Inaccurate or incorrect network parameters;
- Wrong information from the topology processor resulting in possibly large errors.

Therefore, the solution obtained by the real-time state estimator has important influences in functions like the contingency analysis, OPF, dispatcher training simulator, as well as in the new functions needed by the emerging electricity markets. Also, if the input measurement data redundancy is large, the resulting solution should be better.

As described in [17], a state estimation model can be mathematically formulated, as a constrained optimization problem. Thus, by the knowledge that the measurement is equal to the real value plus an error, the following equation can be written:

$$Z = h(x) + e \quad (2.1)$$

Where:

Z - measurements vector;

x - state variables vector;

$h(x)$ - vector function that relates measurements to state variables ("real values");

e - measurement errors (residual) vector.

Since the objective of state estimation is to minimize the measurement errors (e), one can formulate the following:

$$\begin{aligned} &\text{Minimize} && f(Z - h(x)) \\ &\text{subject to} && c(x) = 0 \\ &&& g(x) \leq 0 \end{aligned} \quad (2.2)$$

Where:

$c(x)$ - equality constraint functions vector, used for power flow equations;

$g(x)$ - inequality constraint functions vector, used to define limits for variables;

f - objective function that is being minimized.

The equality constraints mentioned above are used to represent the target values included in the model as pseudo-measurements. On the other hand, the inequality constraints are usually used to represent the limits in the unobservable parts of the network.

Figure 2.3, below, illustrates the inputs and outputs of the state estimator algorithm.

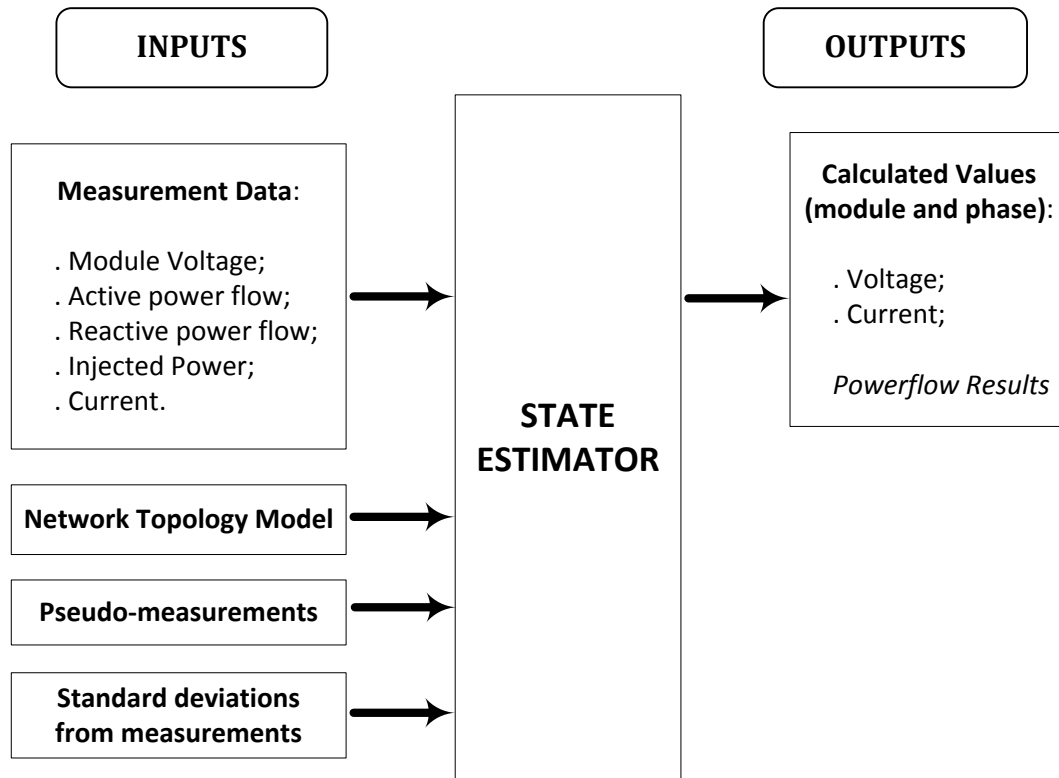


Figure 2.3 - State Estimator diagram.

2.2.1 DC State Estimation using WLS (MSE)

Sometimes it is helpful to work with a simplified approximation of the AC model's non-linear equations. In a DC model some assumptions are made in order to simplify the models and therefore linearize the equations:

- It's assumed that $X \gg R$, a condition typical of large transmission aerial power lines, which makes the resistances approximately zero when compared to the reactances, thus $R = 0$;
- Shunt elements are neglected;
- The node voltages are considered known and constant, because there are no active power losses, therefore are considered to be 1 per unit;
- The angle variation is assumed very small between adjacent nodes, which simplifies the equations to $\cos \theta \approx 1$ and $\sin \theta \approx \theta$.

In literature [14], the formulation is explained in more detail. Therefore, the following equation is presented

$$P = B\theta \quad (2.3)$$

Where:

P - vector of injected active powers;

B - nodal admittance matrix built from the system branch reactances;

θ - vector of nodal voltage angles.

A node has to be selected as reference (slack) and its voltage angle set to zero. This allows the reduction of the equation set by one equation and one variable, and consequently its mathematical resolution. Furthermore, the real power flow from bus i to k (F_{ik}) in a line with a reactance X_{ik} can be approximated by the first order Taylor expansion around $\theta = 0$ and is given by:

$$F_{ik} = \frac{1}{X_{ik}} (\theta_i - \theta_k) \quad (2.4)$$

Since all the power flow equations are linearized, the errors in the DC model state estimation are given by

$$\varepsilon = Z - \hat{Z} = Z - M\hat{\theta} \quad (2.5)$$

Where:

Z - Vector of measurements from the SCADA;

\hat{Z} - Vector of calculated values;

M - Matrix of equation coefficients;

$\hat{\theta}$ - Vector of estimated nodal voltage angles.

In traditional state estimation using the MSE^7 criterion, which corresponds to the application of the classical Least Squares, the estimate is given by:

$$\hat{x} = (H^T \cdot W \cdot H)^{-1} H^T \cdot W \cdot Z \quad (2.6)$$

Since the DC model coefficient equations vector H is linear and constant, and by assuming that the measurement errors have equal weights, the estimated parameter function can be simplified, thus the linear regression expression results in:

$$\hat{\theta} = (M^T M)^{-1} M^T Z \quad (2.7)$$

Because of the assumptions made, this results in (2.7) giving a direct solution vector $\hat{\theta}$, thus not requiring the application of an iterative algorithm.

In conclusion, a simplified WLS regression estimate equation like (2.7) allows the formulation of the state estimation problem in a simple Excel worksheet, and is sufficient, as will be observed in Chapter 4.

⁷ Can also be referred as the Weighted Least Squares (WLS).

2.2.2 AC State Estimation using WLS (MSE)

The traditional AC state estimation algorithm is a very powerful tool in the ability by the system operators to get a real-time (every 1-2 seconds) snapshot of their entire area of control. Using an AC model gives much more accurate results than with a DC model. The most common and traditional state estimation algorithm used is based on the maximum likelihood regression estimator of the Weighted Least Squares (WLS). The generalized state estimation can be solved by the optimization formulation presented in (2.2).

In order to obtain the best estimated values for the module and phase on all node voltages of the power system, having, as input, the available measurement data (with its associated errors) and also the network model (network parameters and topology).

Considering the non-linear measurement model, with m measurements and n state variables, one gets:

$$Z = h(x) + e \Leftrightarrow \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_m \end{bmatrix} = \begin{bmatrix} h_1(x_1, x_2, \dots, x_n) \\ h_2(x_1, x_2, \dots, x_n) \\ \vdots \\ h_m(x_1, x_2, \dots, x_n) \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_m \end{bmatrix} \quad (2.8)$$

Where the variables have similar meaning as presented in (2.1). Additionally:

. x represents:

- V_i ; $i = 1, \dots, N$ (voltage module on all N system nodes);
- θ_i ; $i = 1, \dots, N - 1$ (voltage angle on all N system nodes, except for the slack bus).

. z represents:

- P_{ik} ; Q_{ik} (power flow of the branches (lines or transformers));
- P_i ; Q_i (injected powers on the N busbars);
- V_i (node voltage module).

. $h(x)$ is obtained using the following functions:

a) **Injected power in node i :**

$$\begin{cases} P_i = \sum_{k=1}^N V_i \cdot V_k \cdot (G_{ik} \cos \theta_{ik} + B_{ik} \sin \theta_{ik}) \\ Q_i = \sum_{k=1}^N V_i \cdot V_k \cdot (G_{ik} \sin \theta_{ik} - B_{ik} \cos \theta_{ik}) \end{cases} \quad (2.9)$$

where G_{ik} and B_{ik} represent, respectively, the real and imaginary component of the nodal admittance matrix Y .

b) Power flows from nodes ik and ki:

$$\begin{aligned} P_{ik} &= \frac{R \cdot V_i^2 - R \cdot V_i \cdot V_k \cos \theta_{ik} + X \cdot V_i \cdot V_k \sin \theta_{ik}}{R^2 + X^2} \\ Q_{ik} &= \frac{X \cdot V_i^2 - X \cdot V_i \cdot V_k \cos \theta_{ik} - R \cdot V_i \cdot V_k \sin \theta_{ik}}{R^2 + X^2} - V_i^2 \cdot \frac{\omega C}{2} \\ P_{ki} &= \frac{R \cdot V_k^2 - R \cdot V_i \cdot V_k \cos \theta_{ik} - X \cdot V_i \cdot V_k \sin \theta_{ik}}{R^2 + X^2} \\ Q_{ki} &= \frac{X \cdot V_k^2 - X \cdot V_i \cdot V_k \cos \theta_{ik} + R \cdot V_i \cdot V_k \sin \theta_{ik}}{R^2 + X^2} - V_k^2 \cdot \frac{\omega C}{2} \end{aligned} \quad (2.10)$$

where R and X represent, respectively, the resistance and reactance of the branch (line or transformer).

Furthermore, some commonly used assumptions about the properties of the measurement errors are made:

- a) The average of the errors is zero, i.e. $E(e_i) = 0, i = 1, \dots, m$;
- b) Measurement errors are assumed independent, i.e. $E[e_i e_j] = 0$.

Thus, the $Cov(e) = E[e \cdot e^T] = R = \text{diag}\{\sigma_1^2, \sigma_2^2, \dots, \sigma_m^2\}$

The most commonly used method for solving state estimation using the WLS and, as optimization method the Newton-Raphson (NR), is called “Normal Equations Method”. The NR is very good at solving the quadratic functions, like the objective function previously mentioned (2.2):

$$f(\underbrace{z - h(x)}_{e^T})^T W(\underbrace{z - h(x)}_e) \quad (2.11)$$

where, W represents the weights matrix, which is diagonal and, when using telemetered values, its elements are given by the inverse of the measurement error variances. The NR method will be applied in order to find the zero values from the derivatives of the performance index function $J(x)$, effectively minimizing the following objective function:

$$\begin{aligned} J(x) &= \frac{1}{2} \sum_{i=1}^m (z_i - h_i(x_1, \dots, x_n))^2 \cdot \frac{1}{\sigma_i^2} = [z - h(x)]^T \cdot W \cdot [z - h(x)], \\ \text{with } W &= \text{diag}\left\{\frac{1}{\sigma_1^2}, \dots, \frac{1}{\sigma_m^2}\right\} \end{aligned} \quad (2.12)$$

where σ_i^2 is the (i,i)th element of the covariance matrix.

To find the minimum, the first-order optimal condition for this model can be formulated as:

$$g(x) = \frac{\partial J(x)}{\partial x} = - \sum_{i=1}^m \left(\frac{z_i - h_i(x)}{\sigma_i} \right) \cdot \frac{\partial h_i(x)}{\partial x} = 0 \quad (2.13)$$

However, the minimization of $J(x)$ can't be done in closed form because $h(x)$ is a non-linear function of x . Thus, using a Taylor series expansion, one can approximate the gradient function, as stated in (2.14). Additionally, $G(x)$ is the Hessian matrix of $J(x)$:

$$\begin{aligned} g(x + \Delta x) &\approx g(x) + G(x) \Delta x \\ G(x) &= \frac{\partial^2 J(x)}{\partial x^2} = H^T(x) \cdot W \cdot H(x) - \sum_{i=1}^m \Delta z \frac{\partial^2 h_i(x)}{\partial x^2} \end{aligned} \quad (2.14)$$

Another numerical method named Gauss-Newton (GN), can be used to solve the optimization function, by linearizing the vector $h(x)$, which means neglecting the second (and higher) order terms of the Taylor expansion, thus simplifying the equation, effectively making the original problem a linear least squares problem. However, if the Newton-Raphson numerical method is formulated, using only the first derivatives, both GN and NR have the same iterative expression for updating the state vector. Thus, one obtains:

$$\begin{aligned} G(x) &= H^T(x) \cdot W \cdot H(x) \\ \Delta \hat{x} &= \left[H^T(x) \cdot W \cdot H(x) \right]^{-1} \cdot H^T(x) \cdot W \cdot \underbrace{\left[z - h(x) \right]}_{\Delta z(x)} \end{aligned} \quad (2.15)$$

where, $H(x) = \nabla h(x) = \begin{bmatrix} \frac{\partial h_1(x)}{\partial x_1} & \dots & \frac{\partial h_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial h_m(x)}{\partial x_1} & \dots & \frac{\partial h_m(x)}{\partial x_n} \end{bmatrix}$

This method's algorithm can be summarized in the following steps:

1. Estimate initial values for the state variables x (usually $V_i = 1 p.u.$; $\theta_i = 0 rad$);
2. Iteration k (starts at 1):
 - a. Calculate $h(x)$, $H(x)$ and $\Delta z(x)$ (error deviation);
 - b. Calculate $\Delta \hat{x}_k$;
 - c. Update \hat{x} : $\hat{x}_{k+1} = \hat{x}_k + \Delta \hat{x}_k$;
3. Test convergence: $|\Delta \hat{x}_k| < \text{tolerance}$ (typical value 10^{-4}):
 - If yes: the method ends with $\hat{x} = \hat{x}_{k+1}$; $\hat{z} = h(\hat{x}_{k+1})$ and $J(x) = J(\hat{x})$;
 - If no: go to step 2 and repeat, using $k = k + 1$.

It should be noted that, the only way for reducing the measurement errors (variance) is by having more measurements than state variables, i.e. $m > n$.

In reference [16], the author states that “the numerical performance of the normal equations approach can be negatively affected by the presence of injection measurements, by the use of large weighting factors, and by the presence of very low impedance branches”.

Note: The algorithm used in the AC model state estimation in Chapter 5 is the described here “Normal Equations method”.

As reported before, the WLS method, applied to the static power system state estimation problems, was introduced by Fred Schweppe in references [5; 6; 7]. But since then, there have been many research advances done in this area, including search of newer alternative methods developed. One of the main reasons for this was aimed at finding more efficient methods to solve large LS problems and minimize the disadvantages of the WLS. The WLS is considered the best maximum likelihood estimator, but only if the error's distribution is Gaussian, and that is not always the case, especially the topology (binary) variables (from switches and breakers). Furthermore it is affected by gross measurements that contaminate the other errors, decisively worsening the final solution, meaning it doesn't have bad data filtering capabilities. So, newer and better “bad data” treating techniques were researched, as can be seen in references [9; 10; 11; 12; 13; 31].

2.3 Bad Data processing

One of the advantages of state estimation algorithms is the ability of detecting, identifying and, if possible, eliminating one or more incorrect measurements that might exist. Measurement errors are common because the sensors have limited precision, and there could be synchronization or communication noise issues. Only if redundant measurements are present the gross errors can be eliminated.

It should be noted that different state estimation algorithms use different bad data removal procedures. However, only the most commonly used bad data technique for the WLS will be focused.

Furthermore, the detection and correction of incorrect measurements is usually only performed at the end of the WLS algorithm, since it uses the SE results (calculated residual).

The most popular bad data detection method is using the Chi-Squared distribution (χ^2).

If one considers the independent n variables set (x_1, x_2, \dots, x_n) where each variable x_i has a normal distribution, thus $x_i \sim N(0,1)$ and enough redundant measurements are present $m > n + 1$.

Because of the assumption stated above, the objective function $J(x)$ follows a Chi-Squared distribution. Thus, one gets:

$$J(x) = \sum_{i=1}^n \left(\frac{\Delta z(x_i)}{\sigma_i} \right)^2 = \sum_{i=1}^n X_i^2 \quad (2.16)$$

$$J \sim \chi_k^2(\alpha)$$

Where k is the number of degrees of freedom and is calculated as $k = m - n$ and α is the significance level (e.g. for 95%, the used value is 0.05).

By doing a hypothesis test:

- If $J > \chi_k^2(\alpha)$: there is, at least, one gross measurement present with a confidence level of $1-\alpha$;
- If $J < \chi_k^2(\alpha)$: one can't affirm that a gross measurement is present with a confidence level of $1-\alpha$;

It should be noted that the Chi-Squared function values (right side) are read from a table representing the distribution, while the $J(x)$ equation values (left side) is obtained by numerically solving (2.16).

In case the first test condition is true (bad data found), one needs to find which measurement is the bad one. This is done by calculating the normalized residue (top equation of (2.17)) for each measurement i . Thus the next equation is used:

$$r_i^N = \frac{r_i}{\sqrt{C_{ii}}} \quad (2.17)$$

$$C = R - H \cdot (H^T \cdot W \cdot H)^{-1} \cdot H^T$$

The C_{ii} components match the elements of the diagonal covariance matrix of the residues calculated using the bottom equation in (2.17). Additionally, R represents the error's covariance matrix ($R = W^{-1}$)

After all normalized residuals are calculated, the one that has the highest value is the one eliminated. After the bad measurement removal, the state estimation is rerun, but now excluding the removed measurement, repeating these tests until no bad measurement is detected anymore, by verifying the condition $J < \chi_k^2(\alpha)$.

Chapter 3

Entropy concepts

This chapter will be used to introduce the concept of entropy, its evolution from the Information Theoretic Learning (ITL), and other related concepts. A brief state of the art description will be done on the mentioned theories.

Finally, a state estimation formulation using the maximum correntropy criterion (MCC) will be done for the DC model, but it is also similar to the one used in the AC model as the fitness function of the EPSO.

3.1 Information Theoretic Learning

In reference [32] a good source of information regarding the Information Theoretic Learning (ITL) is offered.

A “mapper” can be defined as a system in shape of an operational block associating inputs to outputs, thus emulating a complex function whose analytic expression is unknown. Neural networks are one example of a mapper, but in a more general definition, mappers are any system that depends on definition of its internal parameters, i.e. weights. A mapper’s training can supervised and non-supervised.

Figure 3.1 represents the three modules that constitute a mapper. The first is the mapper itself, the second is a training algorithm and the last one is a function that measures the performance of the mapper.

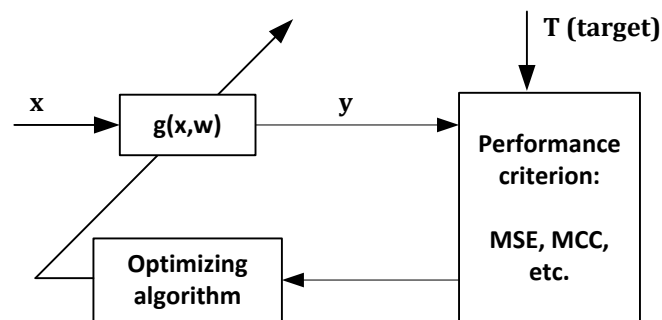


Figure 3.1 - Three module mapper training procedure. [32]

The most common adopted criterion for supervised training has been the minimization of the error's distribution variance when comparing the mappers output (y) and the desired target value (T). The minimization of the variance is more commonly referred as minimum square error (MSE). This criterion's usage has been generalized in almost every optimization algorithms. However the MSE criterion assumes that the error's nature is Gaussian, since it's in the Gaussian distribution that all information is characterized (contained) in only its first two moments (mean and variance). Moreover, by applying the MSE criterion one neglects the higher order moments (e.g. kurtosis, skewness, etc.) of the error distribution. It's easy to acknowledge that in practical cases many error distributions are asymmetrical regarding the mean.

Because of the mentioned limitations of the MSE criterion a new idea emerged in the scientific community, stating that instead of concentrating in the variance, one should instead concentrate in a measurement of the information content of the outputs (in non-supervised training) or error's distribution (in supervised training).

The "information content" mentioned before means a fully and accurate mathematical quantity characterized in Information Theory (IT). The IT is an appropriate way of dealing with the manipulation of information, as introduced by Shannon in two important papers published in 1948-49 [33; 34]. These advances in IT resulted in a huge impact on communication systems, because they provided answers to two main questions: what's the best possible (minimal) code for own data, and what's the maximal quantity of information that can be transmitted through a particular channel.

The measurement of quantity or content of information proposed in IT is called Entropy. And, if according to the IT, entropy measures the information content in a distribution, then it's apparent that, in a supervised training, one needs to obtain the minimization of the error's distribution entropy; while in the non-supervised training, one needs to maximize the output's entropy. As example, for the supervised training case in which the objective was to get an error's distribution with minimum entropy, this error distribution matches a probability density function (pdf) described by a Dirac function (or impulse). If this is verified then one can say that the errors are all the same value, like a mapper's systematic error. An easy way to fix a systematic error is to correct the bias of the exit and one would get a perfect mapper with null error, that perfectly emulates the real (modelled) system. This simple idea couldn't be easily applied because Shannon's Entropy definition wasn't suitable for a practical optimization algorithm. But the Information Theoretic Learning (ITL), a new concept presented by Dr. José Carlos Príncipe in references [35; 36] exploits a different approach of describing the information's entropy, through Renyi's definition which is then combined with Parzen windows methodology (which allows the pdf estimation from a set of discrete samples).

3.2 Parzen Windows

The Parzen window methodology was developed by Emanuel Parzen and is presented in reference [37]. This method can be used to estimate the pdf of a random variable from a discrete sample of points on a M -dimensional space: $y_i \in R^M$, $i = 1, \dots, N$.

This technique uses a kernel function centered in around each point of the sample and looks at that point as being locally described by a Dirac pdf, which is approximated (replaced) by a continuous set whose probability density function is described by the kernel.

Therefore, if a Gaussian kernel (G) is used, the expression of the estimation \hat{f}_y for the real pdf f_y , based on a set of N samples is given by the sum of all the individual contributions of the applied kernel in each point is:

$$\hat{f}_y(z) = \frac{1}{N} \sum_{i=1}^N G(z - y_i, \sigma^2 I) \quad (3.1)$$

where $\sigma^2 I$ represents the covariance matrix (assumed with independent and equal variances in all dimensions). In each dimension k we have the following kernel pdf function centered on the value of point i and with a variance σ^2 :

$$G(z_k - y_{ik}, \sigma^2) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}(z_k - y_{ik})^2} \quad (3.2)$$

Figure 3.2 represents six (discrete samples) Gaussians (in red) and their sum (in blue - stronger), using a variance σ^2 value of 0.5. The Parzen window density estimate \hat{f}_y , is then obtained by dividing this sum by 6, the number of Gaussians. Note that where the points are denser the density estimate will have higher values.

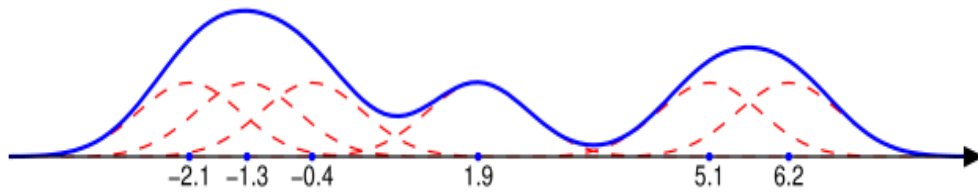


Figure 3.2 - Parzen window density estimate example⁸.

3.3 Renyi's Entropy

Entropy is a concept developed by the Information Theory (IT) that formalizes the content of information notion. The less predictable a message is, the bigger its information content (entropy). Consequently, a message that's completely known has null information content (entropy).

Shannon [33] defined entropy H_S of a discrete probability distribution $P(p_1 \dots p_n)$ as:

$$H_S(P) = \sum_{k=1}^N p_k \log \frac{1}{p_k}, \text{ with } \sum_{k=1}^N p_k = 1 \text{ and } p_k \geq 0 \quad (3.3)$$

This definition had lots of uses, particularly in the communication systems.

⁸ Gathered from http://en.wikipedia.org/wiki/Kernel_density_estimation

However, other entropy definitions exist. For example, Renyi's Entropy formulated in reference [38] is defined as:

$$H_{R\alpha} = \frac{1}{1-\alpha} \log \sum_{k=1}^N p_k^\alpha, \text{ with } \alpha > 0, \alpha \neq 1 \quad (3.4)$$

Renyi's Entropy formulation, actually defines a family of functions $H_{R\alpha}$ dependent on the real valued parameter α . When $\alpha = 2$, we have what's called a quadratic Renyi's entropy:

$$H_{R2} = -\log \sum_{k=1}^N p_k^2 \quad (3.5)$$

This definition can be generalized for a continuous random variable Y with a pdf $f_Y(z)$:

$$H_{R2} = -\log \int_{-\infty}^{+\infty} f_Y^2(z) dz \quad (3.6)$$

This leads to the conclusion that Renyi's Entropy, by containing a logarithm of a sum of probabilities, turned out to be a much more practical algorithm than Shannon's entropy definition, which involves the weighted sum of probability's logarithms. The latter is very hard to work mathematically, in optimization problems while being successfully used in communication systems.

3.4 Correntropy

In references [39; 40; 41] the authors propose and experiment on a new reproducing kernel for stochastic processes that is able to extract information about its time structure, as well as about its statistical distribution.

They defined a generalized correlation function (GCF) in terms of inner products of vectors in a kernel feature space. The authors named this new function as correntropy, because it was deducted that GCF is directly related to Renyi's quadratic entropy estimate of data using Parzen windowing.

Furthermore, it is shown that correntropy exhibits some properties that make it similar in terms of structure to the correlation function. And, unlike the autocorrelation function, the correntropy function transmits information about the Renyi's quadratic entropy of the generating source (i.e., about its moments of higher order).

By comparing WLS with correntropy, the authors concluded that correntropy is local whereas WLS is global. By global, one means that all the samples in the joint space will contribute substantially to the value of the similarity measure while the locality of correntropy means that the value is primarily dictated by the kernel function.

Moreover, the correntropy function has some interesting properties: for example it is a symmetrical positive definite function having a maximum at zero lag, and its mean value is the information potential.

It has been proven that the correntropy function is related to a distance measure $CIM(X,Y)$, between two arbitrary scalar random variables X and Y , also satisfying all the properties of a metric. CIM effectively divides the space in three regions. When the error is small (close to zero), CIM behaves similarly to a L2 metric (Euclidian norm; similar to the

WLS) effectively minimizing the variance. As the error grows (gets farther from zero), CIM behaves like a L1 norm (sum of the differences of coordinates). And, when the errors are very large, CIM becomes a L0 norm, meaning that the metric saturates, thus becoming very insensitive to the gross errors (outliers).

Therefore, it's apparent the importance of the definition of the kernel bandwidth (Parzen window size), because a small kernel size will lead to a small L2 (Euclidean) region, while a larger kernel size leads to a larger L2 region, up to a point where the metric starts behaving like the MSE criterion, but never worse.

Finally, one way of training mappers is using a Correntropy performance function, the maximum correntropy criterion (MCC). Thus, for a Gaussian kernel function with a Parzen window size σ the following expression is obtained:

$$C = \frac{1}{N} \sum_{j=1}^N G(\varepsilon_j, \sigma^2) \quad (3.7)$$

where ε_j is the errors vector.

3.5 State Estimation using MCC

The application of the MCC in DC state estimation problem leads to the following objective function that maximizes the function around the zero mean of the error distribution function thus effectively minimizing the mapper's entropy (maximizing its correntropy). That's because these functions tend to approximate a Dirac function around the zero mean. Therefore, the MCC is given by:

$$\text{Max } C = \frac{1}{N} \sum_{j=1}^N \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2\sigma^2}(Z_j - M_j\theta)^2} \quad (3.8)$$

where M_j is the line j of M and N is the number of measurements.

The expression in (3.8) can be simplified, while maintaining equivalence of maximization principles, thus resulting in an easier to optimize/program objective function:

$$\text{Max } CC = \sum_{j=1}^N e^{-\frac{1}{2\sigma^2}(Z_j - M_j\theta)^2} \quad (3.9)$$

The necessary condition to have a maximum is to derivate expression (3.9) and equal it to zero, thus the resulting equation is:

$$\nabla CC = \frac{\partial CC}{\partial \theta} = 0 \Rightarrow \frac{\partial CC}{\partial \theta} = \frac{1}{\sigma^2} \sum_{j=1}^N (Z_j - M_j\theta) B_{ij} e^{-\frac{1}{2\sigma^2}(Z_j - M_j\theta)^2} = 0 \quad (3.10)$$

These sets of equations were used to solve the DC State Estimation (Chapter 4) problem using an iterative algorithm, in which a simple steepest ascent algorithm (gradient method) was chosen.

On the other hand, for Chapter 5, the AC model used as optimization algorithm the EPSO, which doesn't require using the differentiated equation (3.10), therefore the equation used as fitness function was (3.9), where $Z_j - M_j\theta$ is simply replaced by the error deviation calculated in the EPSO.

Chapter 4

Toy DC Examples

This chapter introduces a small DC toy system of 4 nodes where it's demonstrated how power system state estimation, using correntropy, can show very promising results when compared to the classic MSE criterion, including detection of likely bad measurements and a high degree of insensibility to gross errors in measurements. It is also tested how the Parzen window size variation affects the results. In the next few sub-chapters, tests will be done using the seed measurements, introducing gross errors in measurements and analyze the results.

Some of these subjects were explained in higher detail in Chapter 2 and 3, referring to the State of the Art and the Entropy Concepts.

4.1 Toy System

The toy network is represented in Figure 4.1, with all lines having impedance $z = j0.1 \text{ p.u.}$ and there's an intentional mismatch in generation (0.69 p.u.) and load (0.65 p.u.) measurements. It is also assumed that all measurement errors have equal weights.

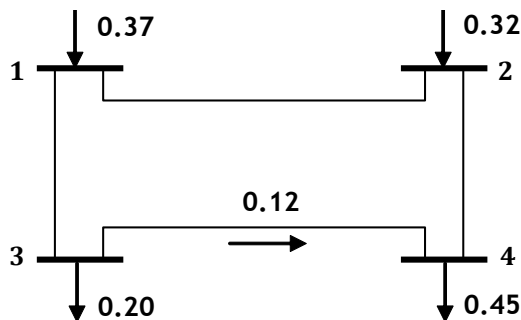


Figure 4.1 - Measured values of active power flows in busbars and in line 3-4 of the toy network.

Figure 4.1 displays the values of power measurements at generation nodes 1 and 2, and load nodes 3 and 4, including an active power flow measurement in line 3-4.

For the DC tests a non-optimized, simple steepest ascent iterative algorithm was used, since the DC network has a very low amount of variables and complexity, thus it being sufficient to successfully converge to an acceptable solution. The step h was adjusted, as needed, to guarantee a quick and most accurate convergence.

There was a need to try several Parzen window size values (σ) for optimum results, as large values of σ would end up giving similar results as WLS and very low values would affect the convergence and quality of the results. Different values of σ were used in the examples, as needed, to get the algorithm to converge to a feasible solution. The variation in window size also allows the inclusion of more or less errors by widening or narrowing the variance around zero.

The system's nodal admittance matrix B is represented by:

$$B = \begin{bmatrix} 20 & -10 & -10 & 0 \\ -10 & 20 & 0 & -10 \\ -10 & 0 & 20 & -10 \\ 0 & -10 & -10 & 20 \end{bmatrix} \quad (4.1)$$

and the power flow in line 3-4 is given by the equation: $F_{34} = 10 \cdot (\theta_3 - \theta_4)$.

Node 1 was defined as reference, so $\theta_1 = 0$ therefore, removing the first column of B , one gets the following system of equations for the state estimation process, as described in (2.5):

$$\varepsilon = Z - M\hat{\theta} \Leftrightarrow Z = M\hat{\theta} + \varepsilon \Leftrightarrow \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ 0.12 \end{bmatrix} = \begin{bmatrix} -10 & -10 & 0 \\ 20 & 0 & -10 \\ 0 & 20 & -10 \\ -10 & -10 & 20 \\ 0 & 10 & -10 \end{bmatrix} \begin{bmatrix} \theta_2 \\ \theta_3 \\ \theta_4 \end{bmatrix} + \begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} \quad (4.2)$$

4.2 Example 1: Default Values

This example uses the default values, as stated previously, to verify how both approaches (MSE/MCC) behave in relation to the deviations of the input measurements. In this sub-chapter, two Parzen window sizes were tested to show the influence in the resulting optimization and error compression around zero. It is expected [40] that when using a large Parzen window size, the MCC will give similar results to the MSE criterion.

By applying the MSE criterion using equation (2.7), one gets

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005286 \\ -0.030905 \\ -0.041381 \end{bmatrix} p.u. \quad (4.3)$$

The estimated active power injections and line flow are calculated using equation (2.3) and gives the following result:

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 20 & -10 & -10 & 0 \\ -10 & 20 & 0 & -10 \\ -10 & 0 & 20 & -10 \\ 0 & -10 & -10 & 20 \\ 0 & 0 & 10 & -10 \end{bmatrix} \begin{bmatrix} 0 \\ \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} 0.361905 \\ 0.308095 \\ -0.204286 \\ -0.465714 \\ 0.104762 \end{bmatrix} p.u. \quad (4.4)$$

Thus the errors using MSE criterion become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ 0.12 \end{bmatrix} - \begin{bmatrix} 0.361905 \\ 0.308095 \\ -0.204286 \\ -0.465714 \\ 0.104762 \end{bmatrix} = \begin{bmatrix} 0.008905 \\ 0.011905 \\ -0.004286 \\ -0.015714 \\ 0.015238 \end{bmatrix} p.u. \quad (4.5)$$

4.2.1 Using Parzen window size $\sigma = 0.005$

Using the MCC equation (3.10) with a Parzen window size of $\sigma = 0.005$ and applying the steepest ascent algorithm referred previously, with a step $h = 0.001$ and using, as starting points, the voltage angle solutions obtained in (4.3) with the MSE criterion, we get:

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005595 \\ -0.031528 \\ -0.043249 \end{bmatrix} \quad (4.6)$$

The estimated active power injections and line flow are now

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 20 & -10 & -10 & 0 \\ -10 & 20 & 0 & -10 \\ -10 & 0 & 20 & -10 \\ 0 & -10 & -10 & 20 \\ 0 & 0 & 10 & -10 \end{bmatrix} \begin{bmatrix} 0 \\ \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} 0.371230 \\ 0.320601 \\ -0.198072 \\ -0.493759 \\ 0.117212 \end{bmatrix} p.u. \quad (4.7)$$

Therefore the errors using MCC become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ 0.12 \end{bmatrix} - \begin{bmatrix} 0.371230 \\ 0.320601 \\ -0.198072 \\ -0.493759 \\ 0.117212 \end{bmatrix} = \begin{bmatrix} -0.001230 \\ -0.000601 \\ -0.001928 \\ 0.043759 \\ 0.002788 \end{bmatrix} p.u. \quad (4.8)$$

Comparing both results, in Figure 4.2, one can easily see that all errors were lower when using the MCC criterion, except for the ε_4 . This leads to the conclusion that the MCC isolated ε_4 as a possible outlier and compressed all the other errors closer to zero to minimize their entropy (maximize their correntropy).

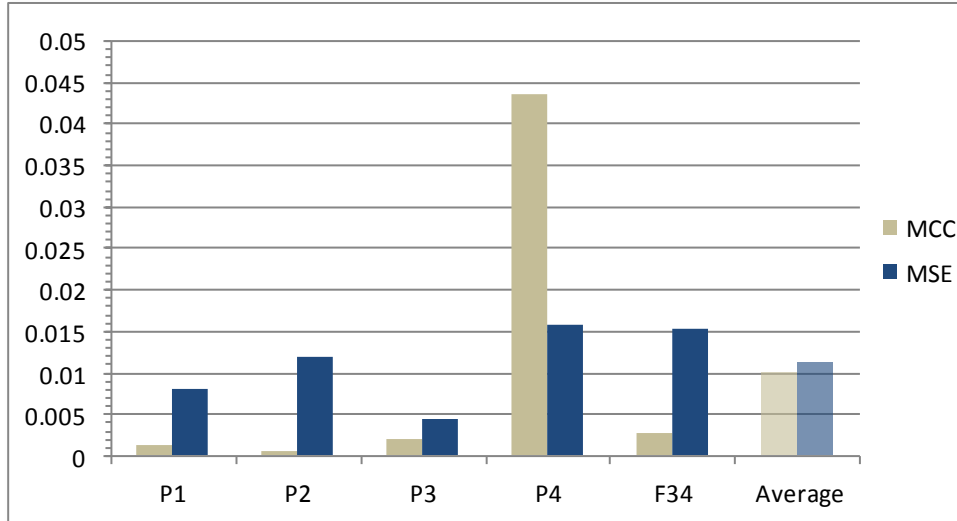


Figure 4.2 - Absolute error comparison⁹ between MSE and MCC in default values and using $\sigma=0.005$.

This behaviour is a direct result of using a small window size that didn't include the error in P_4 as a significant part of the optimization.

Table 4.1 presents the absolute error values' comparison between both studied methods MSE and MCC. It also shows the errors' "Average", that's nothing more than the mean absolute error values.

Table 4.1 - Absolute errors from MSE and MCC, in default values using $\sigma = 0.005$. Red (dark) color in *MCC/MSE Difference* means worse result for MCC when compared to MSE.

	MCC	MSE	MSE/MCC Difference
P_1	0.00123009	0.00809524	-0.00686515
P_2	0.00060104	0.01190476	-0.01130372
P_3	0.00192834	0.00428571	-0.00235737
P_4	0.04375947	0.01571429	0.02804519
F_{34}	0.00278844	0.01523810	-0.01244966
Average	0.01006148	0.01104762	-0.00098614

In Figure 4.3, one can observe the error distribution of both criteria, including the MCC's errors being compressed around zero. The possible bad measurement "detected" by the MCC is further to the right, being isolated as an outlier and not affecting the remaining error's values. Furthermore, in the WLS's error distribution one observes how it's not centered around zero mean which gives a clear idea of the limitations and the wrong assumption (in this case) of the errors having a Gaussian distribution when MSE criterion is used.

⁹ It was decided to employ a widely used and independent method of comparison between MCC and MSE. This method is called Mean Absolute Error (MAE) and is used in both Figure 4.2 like graphs, and Table 4.1 like tables.

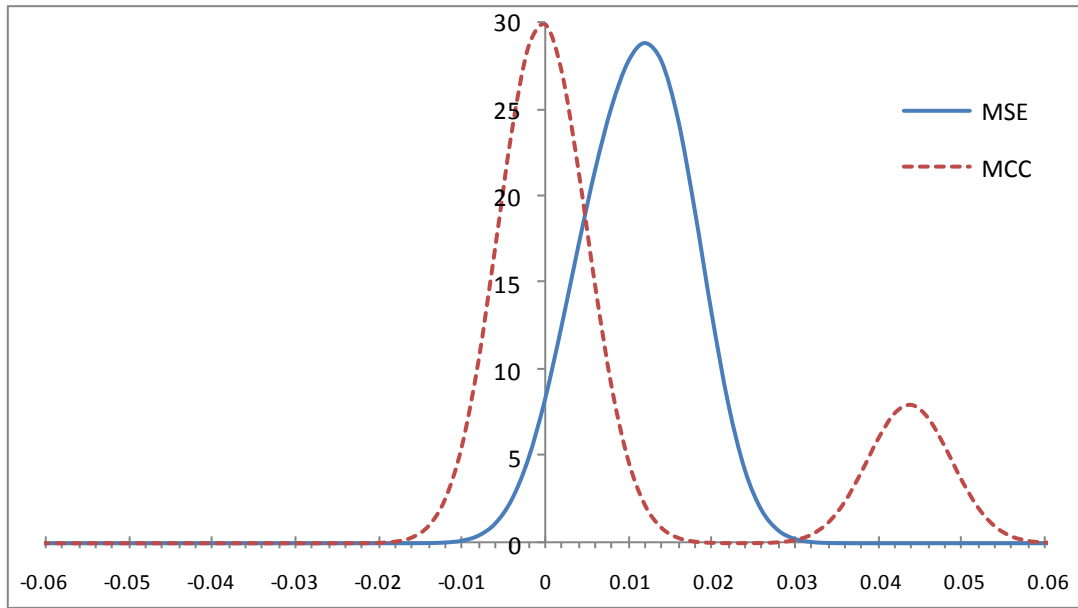


Figure 4.3 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

In Figure 4.4, the same solution as Figure 4.3 is shown but now using a 10 times smaller Parzen window size (σ') to illustrate where the errors are distributed. This is just a change in the graph representation parameter and not a new calculation of the MCC as that would give far different results. It's easily observed that MCC concentrates 4 errors very close to zero and the other one farther to the right. On the other hand, MSE criterion tried to minimize all errors' variance thus contaminating the other errors and worsening the solution.

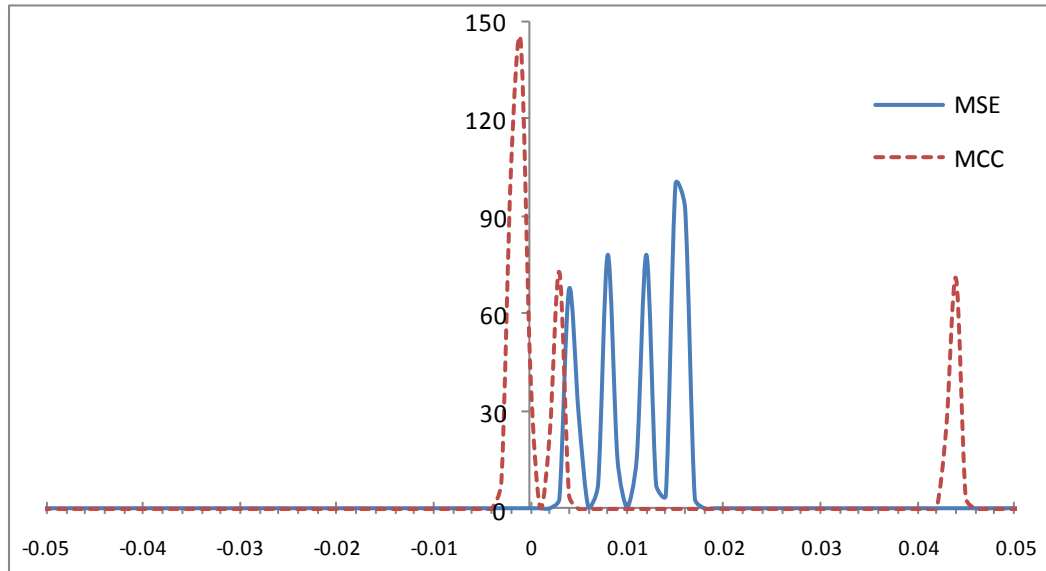


Figure 4.4 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.0005$. The x-axis represents the measurement errors in p.u..

Figure 4.5, below, shows the gradient method evolution, where the solution converged in just under 300 iterations of the maximum 500 defined. It is evident that the outlier is isolated in the first few iterations, and then, the algorithm started compressing the remaining errors.

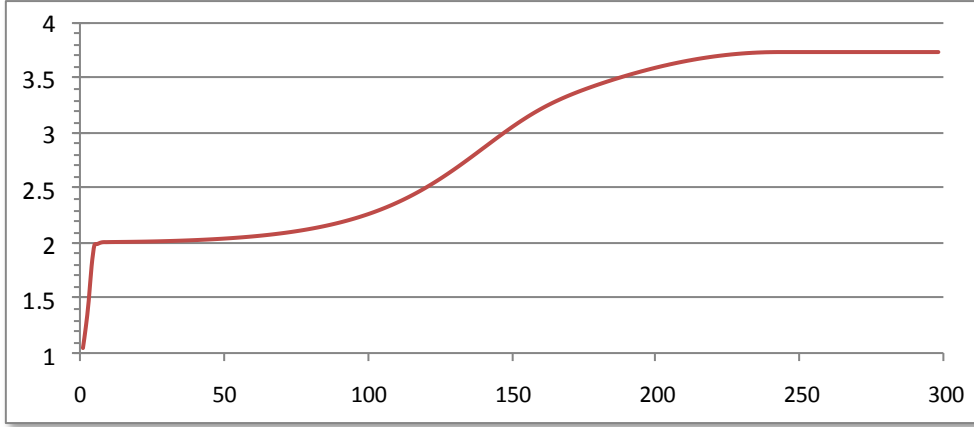


Figure 4.5 - Correntropy evolution curve of the steepest ascent algorithm with $\sigma=0.005$. The x-axis represents the number of iterations and the y-axis is the correntropy value being maximized.

4.2.2 Using Parzen window size $\sigma = 1$

Using now the MCC equation (3.10) with a Parzen window size of $\sigma = 1$, again applying the steepest ascent algorithm with a step $h=0.001$ and using, as starting points, the voltage angle solutions obtained in (4.3) with the MSE criterion, one gets:

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005286 \\ -0.030905 \\ -0.041381 \end{bmatrix} \quad (4.9)$$

The differences between the MCC and MSE are in the 10^{-8} digits and the estimated active power injections and line flow are then

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 20 & -10 & -10 & 0 \\ -10 & 20 & 0 & -10 \\ -10 & 0 & 20 & -10 \\ 0 & -10 & -10 & 20 \\ 0 & 0 & 10 & -10 \end{bmatrix} \begin{bmatrix} 0 \\ \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} 0.361905 \\ 0.308095 \\ -0.204286 \\ -0.465714 \\ 0.104762 \end{bmatrix} p.u. \quad (4.10)$$

Therefore the errors using MCC become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ 0.12 \end{bmatrix} - \begin{bmatrix} 0.361905 \\ 0.308095 \\ -0.204286 \\ -0.465714 \\ 0.104762 \end{bmatrix} = \begin{bmatrix} 0.008905 \\ 0.011905 \\ -0.004286 \\ -0.015714 \\ 0.015238 \end{bmatrix} p.u. \quad (4.11)$$

Observing the results in Figure 4.6, it's verified the already experimented behaviour [40] stating that when a sufficiently large enough Parzen window size is used, the MCC results is similar to the MSE criterion. In other words, it means that the window was large enough to incorporate all the errors and that way, it wasn't able to isolate any outlier, behaving as an L2 metric thus minimizing the errors' variance like MSE.

Table 4.2 presents the calculated measurement absolute errors that are represented graphically in Figure 4.6, including a *MSE/MCC Difference* column that calculates the difference between MCC and MSE values. This can be used to easily compare both methods' behaviour to the large Parzen window value.

Table 4.2 - Absolute errors from MSE and MCC, in default values using $\sigma = 1$. Red (dark) color in *MCC/MSE Difference* means worse result for MCC when compared to MSE.

	MCC	MSE	MSE/MCC Difference
P_1	0.00809498	0.00809524	-0.00000025
P_2	0.01190459	0.01190476	-0.00000017
P_3	0.00428571	0.00428571	0.00000000
P_4	0.01571471	0.01571429	0.00000043
F_{34}	0.01523792	0.01523810	-0.00000017
Average	0.01104758	0.01104762	-0.00000003

Comparing both mean absolute errors, one can clearly see that the differences are minimal as expected. In this case it is in the order of the 10^{-8} digit.

In Figure 4.6, it is presented a visual graph of the previous table values.

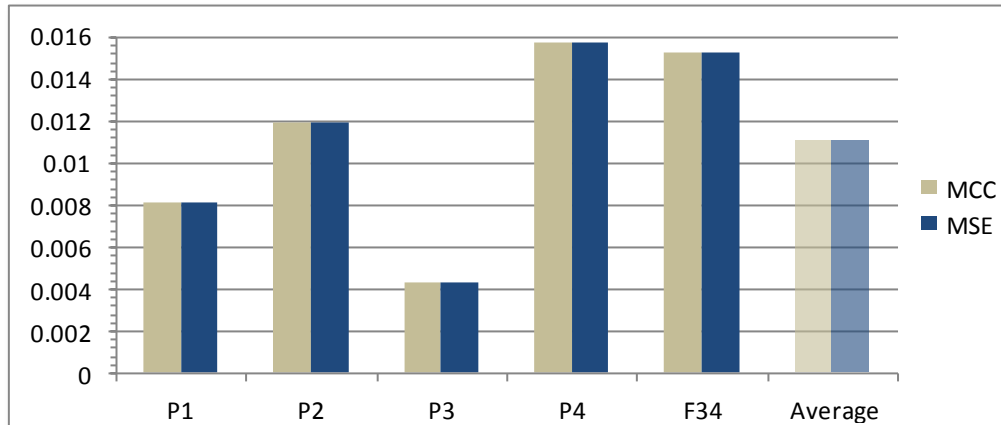


Figure 4.6 - Absolute error comparison between MSE and MCC in default values and using $\sigma=1$.

Figure 4.7 shows both MSE and MCC curves that, in this case, are overlapping each other since the errors are identical and therefore, so is their distribution.

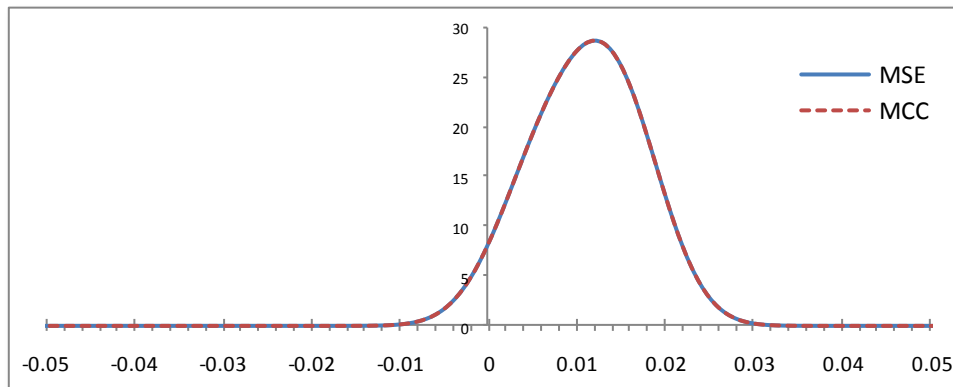


Figure 4.7 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=1$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

4.3 Example 2: $P_4 = -0.6$

To test the effect of using a gross error measurement and the insensitiveness of the MCC, the measured load in node 4 will now be -0.6 instead of -0.45, therefore $P_4 = -0.6$. The other measurements remain the same as before.

The errors in the MCC should be the identical to the ones in the previous test (4.2.1), thus confirming the L2 metric for small errors and L0 metric for detected outliers.

Solving by again applying the **MSE criterion** using equation (2.7), results in:

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.006357 \\ -0.033048 \\ -0.047810 \end{bmatrix} p.u. \quad (4.12)$$

The estimated active power injections and line flow are now

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 0.394048 \\ 0.350952 \\ -0.182857 \\ -0.562143 \\ 0.147619 \end{bmatrix} p.u. \quad (4.13)$$

And the new errors, using **MSE criterion** become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.6 \\ 0.12 \end{bmatrix} - \begin{bmatrix} 0.394048 \\ 0.350952 \\ -0.182857 \\ -0.562143 \\ 0.147619 \end{bmatrix} = \begin{bmatrix} -0.024048 \\ -0.030952 \\ -0.017143 \\ -0.037857 \\ -0.027619 \end{bmatrix} p.u. \quad (4.14)$$

A quick analysis denotes an increase in magnitude of all errors which implies the adverse effect of the deliberately bad measurement (P_4) in all the other measurement errors, leading to the conclusion that the final solution is worse, therefore a filtering process should've been used to remove potential bad measurements before running this algorithm. The problem with the exclusion of likely bad measurements is the eventual lack of redundant measurements, not allowing for the state estimation algorithm to work as intended, therefore behaving like a power flow algorithm instead.

Using the MCC, there was a need to try higher Parzen window sizes to adapt to the larger error variance caused by the introduced gross error, as $\sigma = 0.005$ was too narrow and didn't allow adequate convergence. However, a large window size would also include the influence of the gross error giving a worse solution. The chosen Parzen window size was then $\sigma = 0.007$ since it gave acceptable convergence results. Furthermore, the gradient method step used was $h = 0.001$ and the solution is again calculated the same way.

Like before, the MSE's results are used as initial values of the MCC's gradient method and the resulting voltage angles are:

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005598 \\ -0.031531 \\ -0.043259 \end{bmatrix} p.u. \quad (4.15)$$

The estimated active power injections and line flow are then

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 0.371284 \\ 0.320634 \\ -0.198030 \\ -0.493887 \\ 0.117277 \end{bmatrix} p.u. \quad (4.16)$$

Therefore the new errors using MCC become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.6 \\ 0.12 \end{bmatrix} - \begin{bmatrix} 0.371284 \\ 0.320634 \\ -0.198030 \\ -0.493887 \\ 0.117277 \end{bmatrix} = \begin{bmatrix} -0.001284 \\ -0.000634 \\ -0.001970 \\ -0.106113 \\ -0.002723 \end{bmatrix} p.u. \quad (4.17)$$

Table 4.3 - Absolute errors from MSE and MCC, with $P_4 = -0.6$ using $\sigma = 0.007$. Red (dark) color in MCC/MSE Difference means worse result for MCC when compared to MSE.

	MCC	MSE	MSE/MCC Difference
P_1	0.00128352	0.02404762	-0.02276409
P_2	0.00063366	0.03095238	-0.03031873
P_3	0.00196964	0.01714286	-0.01517322
P_4	0.10611318	0.03785714	0.06825604
F_{34}	0.00272259	0.02761905	-0.02489645
Average	0.02254452	0.02752381	-0.00497929

Like in past results, it is evident, when looking at Table 4.3, that all errors are smaller using MCC except for the detected outlier which is penalized. In other words the correntropy criterion “kicks” the error far away from the zero mean.

The average (MAE) is also smaller, although since it includes the bad measurement the difference isn't that expressive.

If the P_4 error value was removed from the MAE calculations, one would get a difference of **-0.02328812** which is far more revealing of the improvement in the small errors. In Figure 4.8, the error distribution without the influence of the gross error is shown. It gives a great view of the dramatic evolution of all the small errors when comparing MCC to MSE. The

normal error comparison graphic can be consulted in Figure 4.9, where one can see the magnitude difference between MSE and MCC.

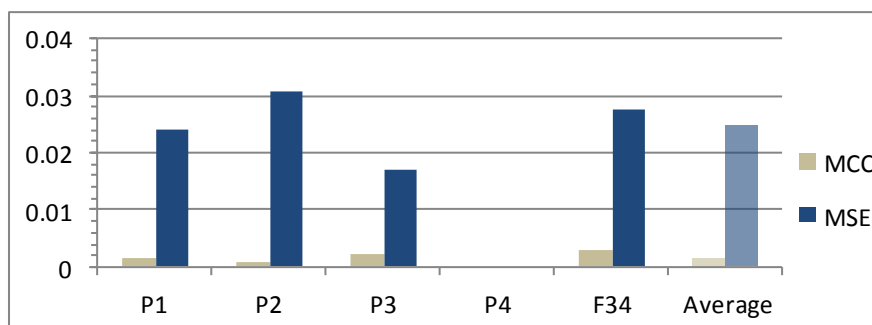


Figure 4.8 - Absolute error comparison between MSE and MCC with $P_4 = -0.6$ and excluding the gross error contribution to the mean absolute error (MAE) calculation.

As reported before and, as expected, the MCC errors are very similar when comparing the example 1 using default values and example 2 using $P_4 = -0.6$. This confirms the MCC's advantage over MSE, where it is able to isolate the gross errors and therefore, avoiding the contamination of the remaining errors. The only error that is very different is the isolated gross error P_4 as expected. For a closer look of the MCC error values of both examples 1 and 2, Table 4.4 can be consulted.

Table 4.4 - Comparison between MCC results in Example 1 and Example 2.

	Example 1 (Default)	Example 2 ($P_4 = -0.6$)
P_1	-0.001230	-0.001284
P_2	-0.000601	-0.000634
P_3	-0.001928	-0.001970
P_4	0.043759	-0.106113
F_{34}	0.002788	0.002723

Thus, we should conclude that even if gross measurements are present to a certain degree, MCC is able to “ignore” those bad measurements causing high entropy, thereby keeping low error values on the remaining measurement errors. This way one avoids the need for separate bad measurement filtering algorithms usually based on statistical approaches.

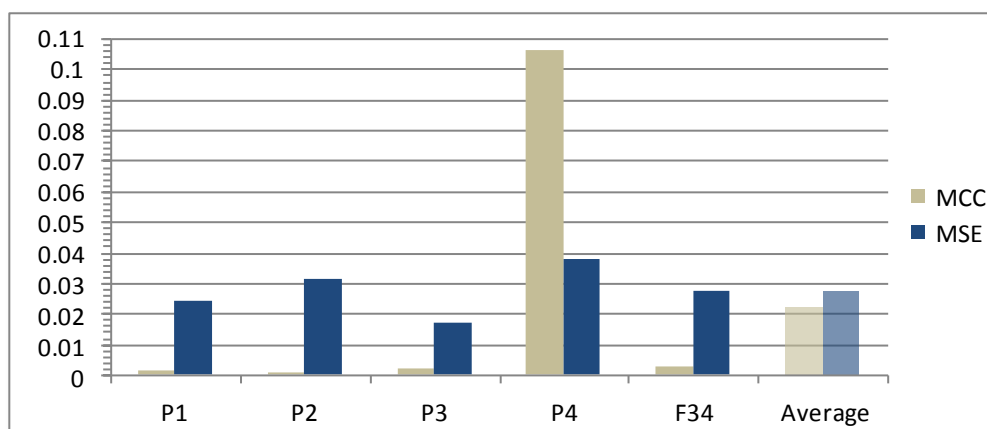


Figure 4.9 - Absolute error comparison between MSE and MCC with $P_4 = -0.6$.

In Figure 4.10, below, just like what happened in Figure 4.3, again is visible; the MCC's errors compressed to zero minimizing their entropy, thus maximizing their correntropy in an attempt to approximate to a Dirac impulse (minimum entropy). Again there's a noticeable "bump" around point -0.10 which is the bad measurement isolated as an outlier, being now farther from the zero mean since the P_4 measurement was worse than before.

The MSE curve is once again not centered on zero mean like example 1 but, it is now farther from the center because all errors increased in magnitude contaminated by the bad measurement introduced in this example.

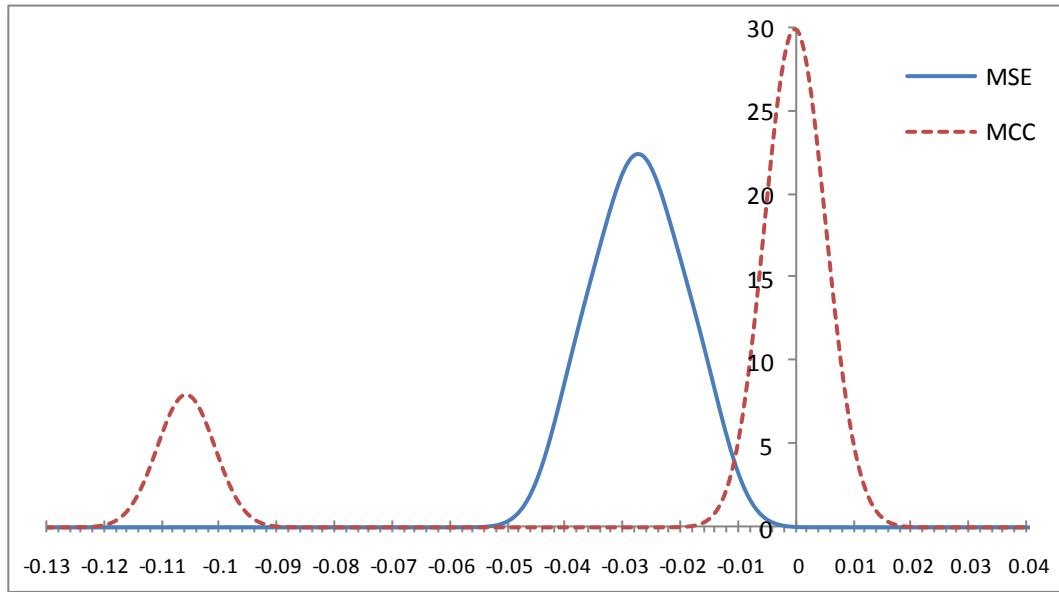


Figure 4.10 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.007$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

It is also noticeable when comparing Figure 4.10 to Figure 4.3 that, while in the first example there was a positive MCC gross error it is now negative. That's simply because the measurement was in deficit, since the optimal value MCC gave was around -0.5. However, in this case the error is in excess since a -0.6 measurement was used.

4.4 Example 3: $F_{34} = -0.12$

This next example uses all default values from the initial problem, except for the active flow in line 3-4 that is now reversed to simulate a somewhat common measurement error from the electric network metering devices.

For convergence to be achieved there was a need to widen the Parzen window to $\sigma=0.05$ so that the simple steepest ascent algorithm didn't get stuck in the several local optimums present. This means that there are smoother curves therefore, less likely for the algorithm to get stuck. Care needs to be taken when increasing too much the window size, since it will start including the influence of the gross measurement error, thus giving similar results to the MSE criterion. That is something to be avoided.

Applying the **MSE criterion** and using equation (2.7), one gets

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.001857 \\ -0.032048 \\ -0.036810 \end{bmatrix} p.u. \quad (4.18)$$

Thus, the estimated active power injections and line flow are

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 0.339048 \\ 0.330952 \\ -0.272857 \\ -0.397143 \\ 0.047619 \end{bmatrix} p.u. \quad (4.19)$$

And the new errors, using **MSE criterion** are calculated as

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ -0.12 \end{bmatrix} - \begin{bmatrix} 0.339048 \\ 0.330952 \\ -0.272857 \\ -0.397143 \\ 0.047619 \end{bmatrix} = \begin{bmatrix} -0.030952 \\ -0.010952 \\ -0.072857 \\ -0.052857 \\ -0.167619 \end{bmatrix} p.u. \quad (4.20)$$

Just like in previous examples, there's an increase in all errors which again confirms that the MSE criterion affects all errors, by trying to minimize the error's variance, thus ultimately contaminating the gross error's variance to all the remaining measurement errors. As a result, it's apparent once again that MSE is not good when in presence of bad measurements. Therefore it would require a filtering algorithm to remove the gross errors before the MSE algorithm is executed.

Using the **MCC criterion** with a Parzen window size of $\sigma=0.05$ and applying the steepest ascent algorithm again, with a step $h = 0.001$, using as starting points the voltage angle solutions obtained above with the MSE criterion, one gets

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005000 \\ -0.031000 \\ -0.041000 \end{bmatrix} p.u. \quad (4.21)$$

Thus the estimated active power injections and line flow are now

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 0.359998 \\ 0.310002 \\ -0.210005 \\ -0.459995 \\ 0.099995 \end{bmatrix} p.u. \quad (4.22)$$

Therefore, the new errors using **MCC criterion** become

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ -0.12 \end{bmatrix} - \begin{bmatrix} 0.359998 \\ 0.310002 \\ -0.210005 \\ -0.459995 \\ 0.099995 \end{bmatrix} = \begin{bmatrix} 0.010002 \\ 0.009998 \\ 0.010005 \\ 0.009995 \\ -0.219995 \end{bmatrix} p.u. \quad (4.23)$$

The results obtained with MCC in this case are very good, since it isolated the inverted measurement and used, as the most coherent solution, the value of 0.10p.u. which is very close to the original measurement value of 0.12p.u.. Moreover, the remaining errors were treated by MCC like if a L2 metric algorithm (MSE criterion) was used, dividing the errors' variance while minimizing by all remaining measurements, resulting in all of them having a value close to -0.01p.u.. To strengthen this notion further, in sub-section 4.5 a new test is done using the *exact*¹⁰ measurement value for F₃₄.

Table 4.5 - Absolute errors from MSE and MCC, with F₃₄ = -0.12 (inverted) using $\sigma = 0.05$. Red (dark) color in *MCC/MSE Difference* means worse result for MCC when compared to MSE.

	MCC	MSE	MSE/MCC Difference
P ₁	0.01000183	0.03095238	-0.02095055
P ₂	0.00999817	0.01095238	-0.00095421
P ₃	0.01000548	0.07285714	-0.06285166
P ₄	0.00999451	0.05285714	-0.04286263
F ₃₄	0.21999543	0.16761905	0.05237638
Average	0.05199909	0.06704762	-0.01504853

Again, it is obvious, when looking at Table 4.5, that all errors are smaller using MCC except for the detected outlier which is penalized. The average (MAE) is also smaller however, since it includes the bad measurement, the difference isn't that significant.

If the F₃₄ error value was removed from the MAE calculations, the new obtained difference is **-0.03190476** which is a far more enlightening improvement over the small errors. Figure 4.11, below, shows the error comparison graph without the influence of the gross error in the MAE calculation. There is a huge evolution on all the small errors when comparing MCC to MSE.

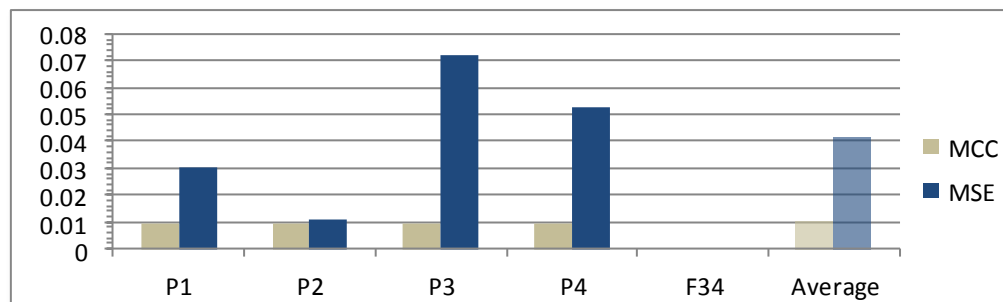


Figure 4.11 - Absolute error comparison between MSE and MCC with F₃₄ = -0.12 (inverted) and excluding the gross error contribution to the mean absolute error (MAE) calculation.

¹⁰ Value obtained in the example 3 by the MCC, of 0.10p.u.

It should be noted, however, that the solution obtained here is clearly different than the one in previous solutions (small error values). A comparison is shown in Table 4.6, and one can conclude that, as previously reported, there are many feasible solutions and it can't be said that the obtained solution is the best one available. This is also explained because of using a different measurement as gross error (F_{34}). Different errors can affect the network in different ways, hence the alternative solutions.

It can be safely assumed, nevertheless, that the obtained solution is one of the optimum solutions and is, certainly, a great improvement over the MSE criterion results.

Table 4.6 - Comparison between MCC results in Example 1 ($\sigma = 0.005$), 2 ($\sigma = 0.007$) and 3 ($\sigma = 0.05$).

	Example 1 (Default)	Example 2 ($P_4 = -0.6$)	Example 3 ($F_{34} = -0.12$)
P_1	-0.001230	-0.001284	0.010002
P_2	-0.000601	-0.000634	0.009998
P_3	-0.001928	-0.001970	0.010005
P_4	0.043759	-0.106113	0.009995
F_{34}	0.002788	0.002723	-0.219995

In Table 4.6, the results lead to the conclusion that for example 1 and 2, MCC concentrated the error on measurement P_4 while minimizing much more the remaining error's variance.

Comparing results in Figure 4.12, we again observe the same behaviour as in example 1 and 2, only this time there's a higher increase in the MSE criterion's measurement errors while the MCC errors continue to be very low except for the isolated error F_{34} .

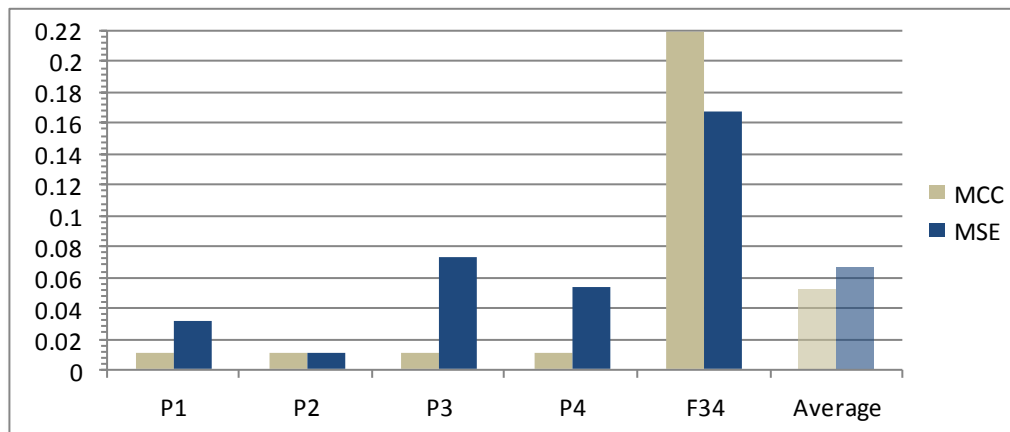


Figure 4.12 - Absolute error comparison between MSE and MCC with $F_{34} = -0.12$.

To better illustrate how the differences in the Parzen window size dramatically change the shape of the curves, Figure 4.13 and Figure 4.14 are presented below. Once again, the small window size leads to a spikier curve and, on the other hand, a larger window size gives a much smoother curve. It is very hard to optimize a correntropy function with very small Parzen window size, since it gives a large number of local optimums and, at least an optimization algorithm such as gradient method gets stuck in the peaks.

Figure 4.13 shows the MCC compressing the errors around zero and isolating the bad measurement (F_{34}), just like in the other cases showing how robust it can be, even with a weak optimization algorithm.

The behaviour of the MSE criterion is different because the errors are now much more affected by the bad measurement and therefore resulting in them being more separate from each other, thus having a higher variance. It's safe to assume that the MSE criterion contaminated the other errors in a decisive way and was unable to effectively correct them.

An inverted measurement usually represents aggressive change in network power flows, therefore this is a perfect example to experiment on. The obtained results show the strengths of the maximum correntropy criterion (MCC) over the minimum square error (MSE).

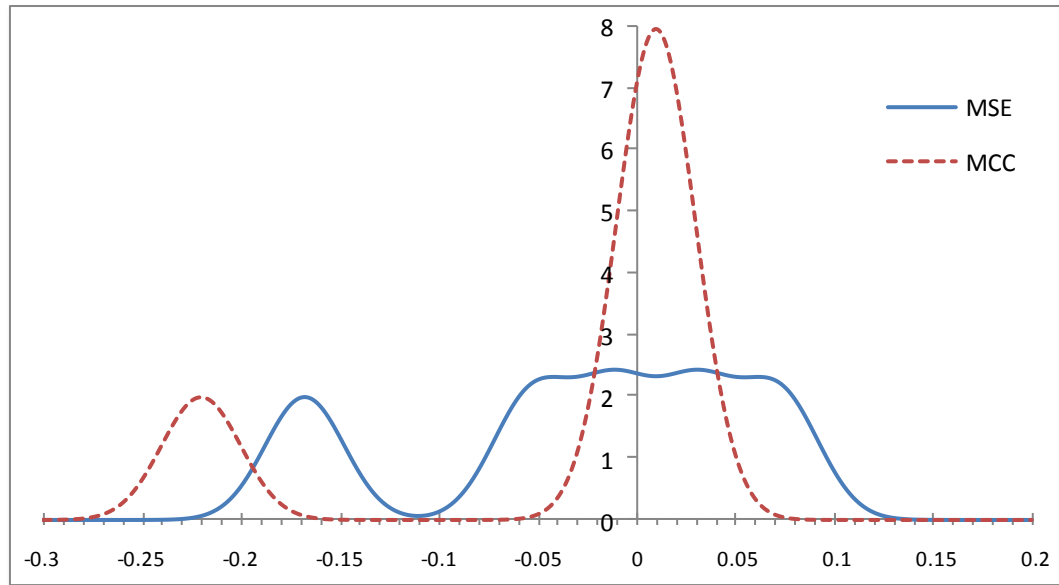


Figure 4.13 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.02$. The x-axis represents the measurement errors in p.u..

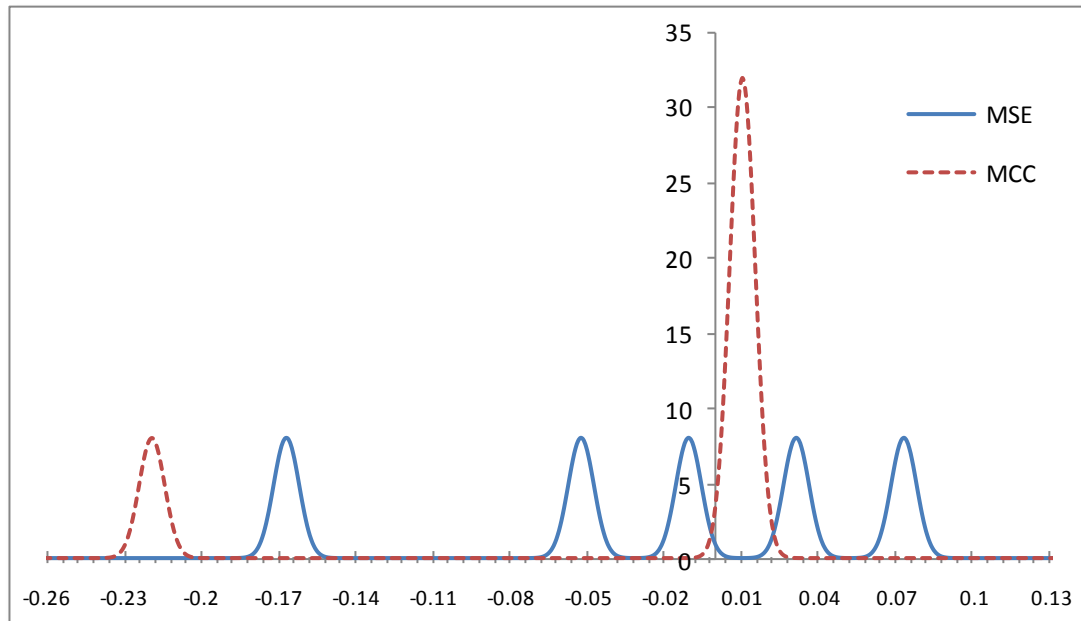


Figure 4.14 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

In Figure 4.14, it's noticeable how the 5 errors from the MSE criterion were spread apart, while with the MCC only one error was isolated from the rest and all the others were very close to each other and close to the zero mean, more precisely, around 0.01. The MCC provided much better results overall, and there wasn't a need to eliminate the bad measurement.

The obtained results are very promising, since investigators have been researching for new computational ways to test the measurement errors and eliminate them as a separate process. For example, a different approach is used in the thesis "*A State Estimation Approach for Distribution Networks Considering Uncertainties and Switching*" [16] the developed RTSE application¹¹, among several error correction techniques used, one was simply testing if a line measurement was reversed by trying one run of the estimator; if it didn't converge and some conditions were met it tried reversing the measurement and re-run the estimator algorithm. Most gross error removal techniques rely on probability distributions such as the Chi-Square (χ^2). This obviously adds complexity to the program and slows the execution time of the RTSE software.

From the tests performed in all three previous examples, it's safe to assert that it's not needed to run the corrective algorithms since MCC simply isolates the gross errors and keeps optimizing the other ones.

4.5 Example 4: $F_{34} = 0.10$

This is the last example on the DC model and what will be tested is how the MCC behaves if the gross measurement was simply eliminated, that is, the "exact" value (calculated in example 3) was used as measurement. Following the same reasoning about MCC, as introduced and experimented on [40], it is expected that if no gross errors are present the MCC should behave as a L2 metric, thus it's optimization evolution is just minimizing the error's variance like MSE criterion does. This effect is obviously only verified if the Parzen window size is the same in both tests as a variation (lowering the value) would cause one of the errors to be detected as an outlier and thus it would try to minimize it's entropy (maximize the correntropy). In Figure 4.16 and Figure 4.17, and just as illustration of the previous sentence, the graphs with the corrections are shown using a Parzen window size $\sigma = 0.005$ (10x smaller) as optimization parameter. This is just one of the many particularities that need to be taken into account when dealing with MCC. However this Thesis's goal is mainly to prove the concept regarding the existence of other (better) solutions to the state estimation process when using correntropy.

Therefore, using the same measurements as before (with the exception of F_{34}), which now has a value of 0.10 the MSE values are being calculated.

¹¹ Real-Time State Estimator application developed by Prof. Jorge Correia Pereira at INESC Porto

Again applying the **MSE criterion** by using equation (2.7), one gets

$$\begin{bmatrix} \hat{\theta}_2 \\ \hat{\theta}_3 \\ \hat{\theta}_4 \end{bmatrix} = \begin{bmatrix} -0.005000 \\ -0.031000 \\ -0.041000 \end{bmatrix} p.u. \quad (4.24)$$

Thus, the estimated active power injections and line flow are now

$$\begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ P_4 \\ F_{34} \end{bmatrix} = \begin{bmatrix} 0.360000 \\ 0.310000 \\ -0.210000 \\ -0.460000 \\ 0.100000 \end{bmatrix} p.u. \quad (4.25)$$

And the new errors, using **MSE criterion** are calculated as

$$\begin{bmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \varepsilon_3 \\ \varepsilon_4 \\ \varepsilon_5 \end{bmatrix} = \begin{bmatrix} 0.37 \\ 0.32 \\ -0.2 \\ -0.45 \\ 0.1 \end{bmatrix} - \begin{bmatrix} 0.360000 \\ 0.310000 \\ -0.210000 \\ -0.460000 \\ 0.100000 \end{bmatrix} = \begin{bmatrix} 0.010000 \\ 0.010000 \\ 0.010000 \\ 0.010000 \\ 0.000000 \end{bmatrix} p.u. \quad (4.26)$$

If the results in (4.26) with (4.23) are compared, one easily verifies that all errors are identical, except for the gross error introduced in example 3 (exact value in example 4). To facilitate the comparison, Table 4.7 gives errors side by side.

Table 4.7 - Error comparison of MCC in example 3 and MSE in example 4, with $F_{34} = 0.10$ ("exact").

	MCC	MSE
	Example 3 ($F_{34} = -0.12$)	Example 4 ($F_{34} = 0.10$)
P_1	0.010002	0.010000
P_2	0.009998	0.010000
P_3	0.010005	0.010000
P_4	0.009995	0.010000
F_{34}	-0.219995	0.000000

Since the MCC criterion gives the exact same results there's no need to repeat the calculations. With the obtained solution it is demonstrated, beyond doubt, the positive behaviour of the MCC when in presence of gross errors and also of only small errors.

It is important to reiterate once more that, this result is only possible when the Parzen window size remains the unchanged in order to treat the errors the same way. Increasing the window size doesn't cause variation in results however, when narrowing the window one of the other errors will end up being caught and detected as a possible outlier thus inevitably changing the solution.

Figure 4.15, below, is just shown out of curiosity to reinforce the idea and illustrate the error's Gaussian distribution, using the same window size as before ($\sigma = 0.05$) as optimization parameter, and $\sigma' = 0.005$ as representation scaling parameter.

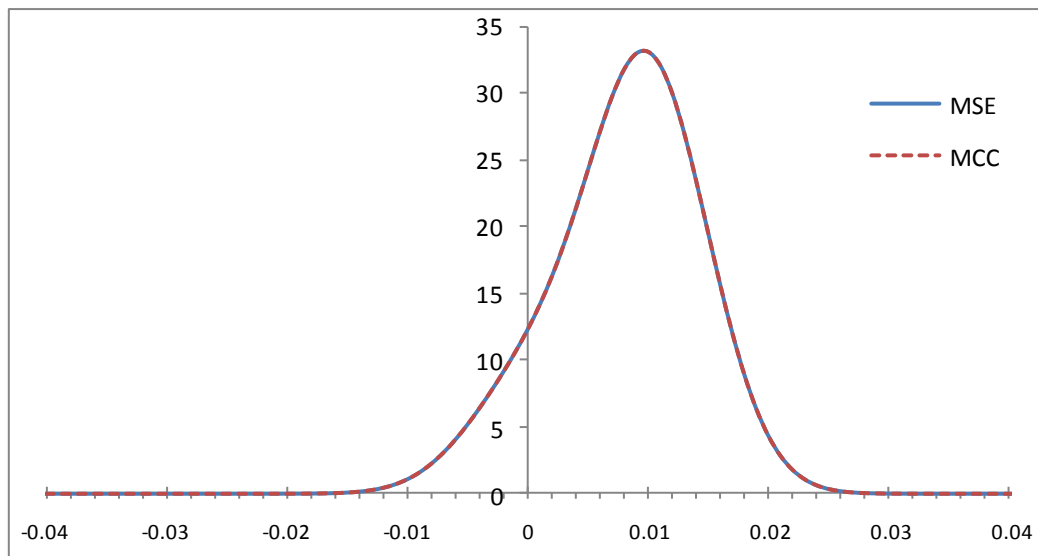


Figure 4.15 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.05$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

In Figure 4.16, once again there is an evolution in the MCC optimization (from 0.05 to 0.005). However it is not an acceptable solution and that could be, in part, due to the lack of redundancy of measurements, or because of the narrow Parzen window size used, which detected one of the small errors as outlier, isolating it.

The algorithm isolated the measurement error of the injected power in node 2 and tried to compress the remaining errors.

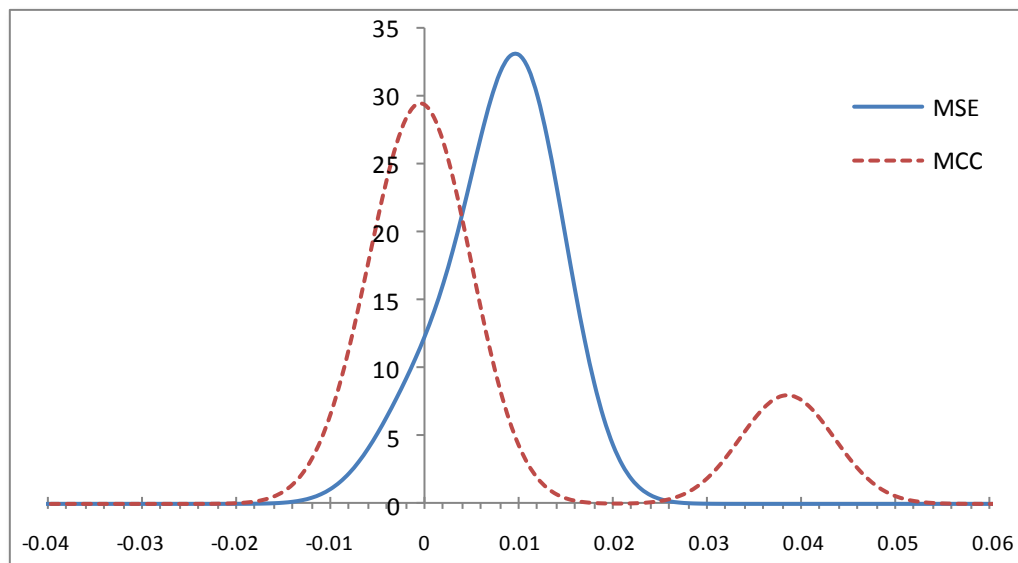


Figure 4.16 - Error distribution of MSE and MCC, using Gaussian Kernel functions with a Parzen window size $\sigma=0.005$ and $\sigma'=0.005$. The x-axis represents the measurement errors in p.u..

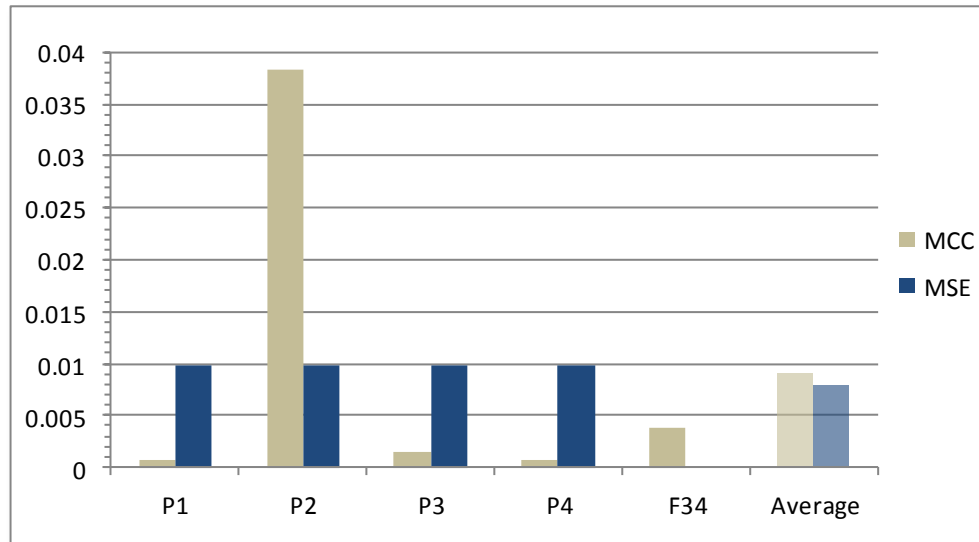


Figure 4.17 - Absolute error comparison between MSE and MCC with $F_{34} = -0.10$ and using a new optimization parameter $\sigma=0.005$, to the mean absolute error (MAE) calculation.

4.6 Chapter conclusion

In conclusion, it's worth mentioning that the DC model testing was successful and the proposed objectives and conclusions were met. Since the DC model is of simple application and the power system network is small, the complexity of the tests was not too great. However, its simplicity was enough to test different measurement configurations and scenarios and that way proving the concept. It's worthy to note that this was accomplished with an un-optimized and simplistic gradient (steepest ascent) method, showing the robustness of the MCC.

It was demonstrated in **example 1** how the maximum correntropy criterion (MCC) behaves in the presence of two extreme Parzen window size values. Thus, large window sizes have the ability of making the MCC behave just like a L2 metric, in other words, tries to minimize the variance of all measurement errors. On the other hand small window sizes make MCC behave like a L2 metric for small errors, L1 metric for intermediate errors and L0 metric to large (gross) errors.

As for **example 2**, it was used to reinforce the L0 metric properties of the MCC by increasing the error value and observing that the MSE criterion increased the error's variance in all the errors, therefore worsening the solution. MCC, in contrast, kept the small measurement error's values similar to the original case (in example 1) thus proving that it is insensible to the gross error variation on the remaining errors.

On **example 3**, a more concrete test was made: inverting the flow in line 3-4. Again the robust correntropy properties were verified and even though MSE criterion gave very poor results the MCC, using a simple gradient method, was able to find a better solution to the problem. The MCC calculated the most coherent value for the flow as 0.10p.u. and for the remaining errors it split the variance between them. In particular, it reduced the generation

by 0.01p.u. on each generator and increased the same value of 0.01p.u. on each load. For the gross measurement it was just kicked away from the zero mean.

Finally, **example 4** was used to demonstrate that MCC gives the exact same solution as MSE (L2 metric), under some conditions (e.g. same Parzen window size), if the gross error is corrected to its “exact” value (calculated by MCC).

It is important to refer that the solutions obtained using MCC can be called optimal solutions but not best solution available, since the gradient method doesn’t allow for broad dimensional space sweep. Also because of the nature of the optimization function there are several local optimums with could make the gradient method stuck.

The next chapter will be used to test a 24-bus AC power system network which is far more complex, thus requiring better optimization algorithms to converge to an optimal solution.

Chapter 5

Solving the AC problem with EPSO

The power system network shown in Figure 5.1, below, is the same as “*IEEE Reliability Test System*” [42] with slight modifications, namely, some loads were removed. Complete information about this network, including generation, loads and line parameters and characteristics can be found in the Annex A.

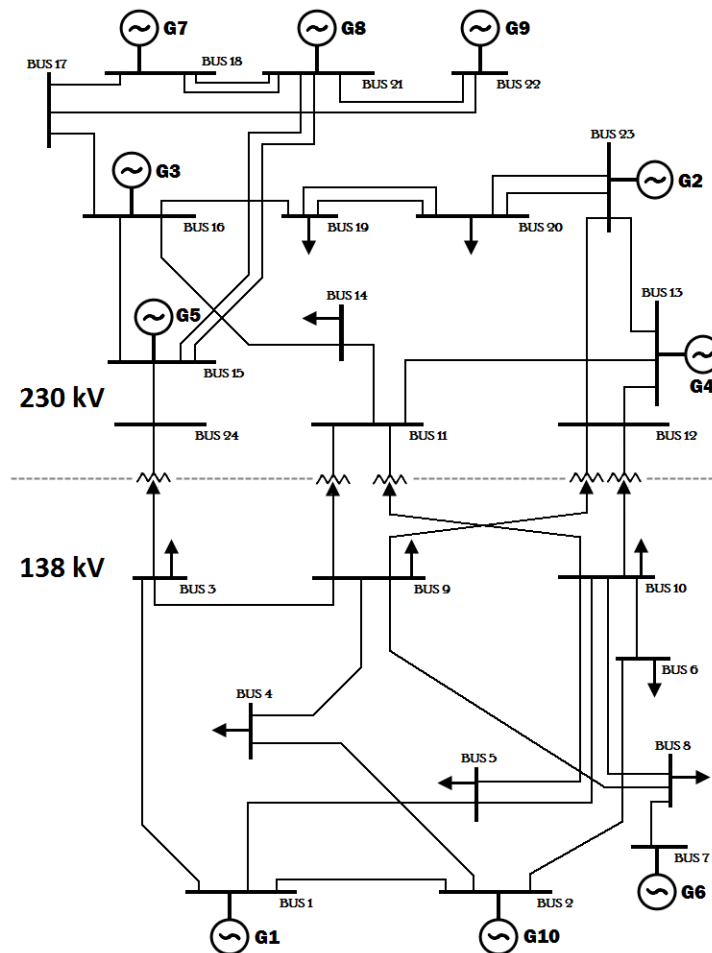


Figure 5.1 - Network topology of the IEEE Reliability network with minor modifications.

In this chapter a full AC network will be tested. The chosen network is a known reliability test system and is used in many power system studies. With this AC model network, a more complex test will be done, but the conclusions ought to be similar to the ones obtained in Chapter 4, with the DC model.

Because of the number of variables and consequent complexity there was a need to use a more powerful optimization algorithm, as the gradient method just wasn't appropriate. Unlike a classical mathematical algorithm, such as the gradient method where, if one starts from the same initial solution, and the same conditions apply (e.g. same step), the exact same solution is reached every time; a meta-heuristic like the EPSO allows for dimensional space sweeping and thus is able to find a larger number of local optimums by varying for example the Parzen window size, and it's not guaranteed that the same solution is always found, which for the kind of analysis and tests made in this Thesis, was adequate.

Another classical and the most common optimization algorithm for state estimation is the Newton-Raphson. These algorithms need, however, that the objective function is differentiable and the starting points are of utmost importance as their values can affect the convergence. Even though the MCC equation (3.8) is differentiable which allows for a future use in the much faster Newton-Raphson, it's out of the scope of this thesis the mathematical deduction and respective application of it.

Therefore, other optimization algorithms are preferred, such as the evolutionary algorithms (EA), since they work as a "black box" and don't require knowledge of the eventually complex mathematical functions. Thus, the chosen algorithm was the Evolutionary Particle Swarm Optimization (EPSO). However, since it is a meta-heuristic and evolutionary algorithm care needs to be taken before applying the optimization, such as, finding the best mutation parameters and cooperation probability. Those variables will be calculated in the next sub-chapters, before starting to solve the test cases. The problem with evolutionary algorithms is the fact that they don't give the same solution every time like the classic ones do. To counter this issue, one needs to do, for example, 10-20 runs of the algorithm and calculate the mean of the solutions to get a more trustworthy result. Therefore, in all the preformed tests in this chapter, 10 runs were done in each configuration and then an average of the results was done.

The Real Time State Estimator (RTSE) program used, was developed at INESC Porto by *Prof. Jorge Pereira* [16; 43; 44], among others, and uses, as optimization criterion, the MSE which is the most common regression model used globally in the state estimation problem.

This way, the challenge was to link the RTSE results with the EPSO and use, as starting points, the MSE error results. Therefore, the EPSO algorithm program, developed also at INESC Porto by *Hrvoje Keko* [45; 46; 47] had to be adapted to use the maximum correntropy criterion's fitness function and the already mentioned mutation and cooperation parameters. There is also the particle population number that had to be defined. Since the EPSO problem dimension was 48 (24-bus system x node voltage (module + angle)), some population sizes (50, 60, 80, 100, 200) were tested and a population of 60 was used. The other (higher) populations slightly accelerated the convergence, but since it wasn't significant and the resulting processing power (execution time) was moderately increased, it was decided to use 60 particles, mainly because of time constraints. More information about the EPSO algorithm formulation and properties can be found in Annex C.

5.1 Determining the communication probability “p”

The communication probability “p” is an externally fixed parameter that controls the passage of information within the swarm. One way to introduce stochastic star topology (Figure 5.2-b) is through the communication factors P among particles that are present in the main EPSO equation (C.6). These communication factors are represented by a diagonal matrix that has binary variables: a value of 1 with a probability p and value of 0 with a probability $(1-p)$. In the classical star communication topology (Figure 5.2-a) the “p” value is equal to 1.

Therefore, for each iteration and search space dimension, there is a probability “p” that a particle will not receive the information available on the best location found by the swarm. Thus, the importance of this parameter is apparent since, if a low communication probability value is used, the particle would almost only evolve under the influence of inertia and memory components, or, in other words, the particle (descendent) is generated almost only through the influence of its parent particles.

It is believed that “restraining the free flow of information about the global best allows more local search by each particle, eliminates disturbing noise, allows the dynamics of particle movement to be more stable and avoids premature convergence” and “It has been shown that a full star communication acts against exploration of the search space and may lead to premature convergence.” and “On the other hand, alternative structures with extremely low communication between particles (...) happen to affect the social component of particle swarm optimization and risk to convert the particle swarm search process into something close to a set of parallel individual searches.” [46].

Thus, if an adequate communication probability is used it will result in more local searches by each particle, eliminates the disturbing noise and allows the particle to evolve in a more stable way, also avoiding a premature convergence, leading obviously to better results.

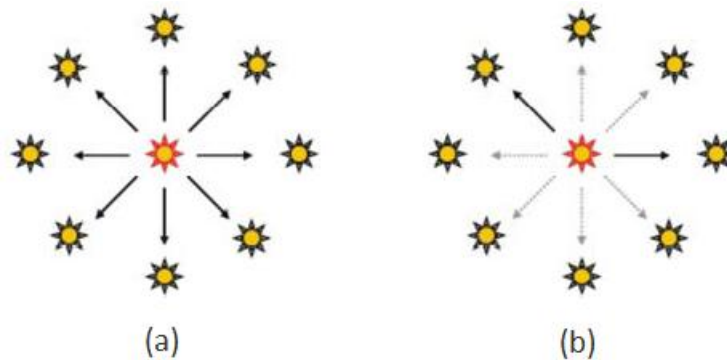


Figure 5.2 - Illustration of the star communication topology (a) and the stochastic star topology (b) [46]

To determine the optimal value for the communication probability “p”, two tests were done for different values of “p”, using a constant Parzen window value $\sigma=0.5$ and $\sigma=0.2$, and learning rate (mutation parameter) $\tau=0.3$ and $\tau=0.1$. The tests used the 24-bus system described in Figure 5.1 and Annex A, except for a +5% error introduced in the active power of line 5-10.

All tests' mean absolute error (MAE) values were calculated in a similar way as in Chapter 4 for the DC model. The total number of fitness function evaluations is 720000 for the 60 particles, and 6000 iterations used as the stopping criterion. Table 5.1 lists the MAE values for each test, as function of the cooperation probability, where lower values are better.

Table 5.1 - Cooperation probability “p” tests (best value in gray). For reference, the mean absolute error (MAE) value for MSE is 0.090086.

Learning rate (τ) / Parzen window size (σ)			
Coop. Probability		0.1 / 0.2	0.3 / 0.5
	0.0	0.08751	0.08885
	0.2	0.04006	0.04203
	0.4	0.03808	0.04228
	0.6	0.03792	0.03849
	0.7	0.03947	0.03886
	0.8	0.03972	0.03890
	1.0	0.08204	0.08272

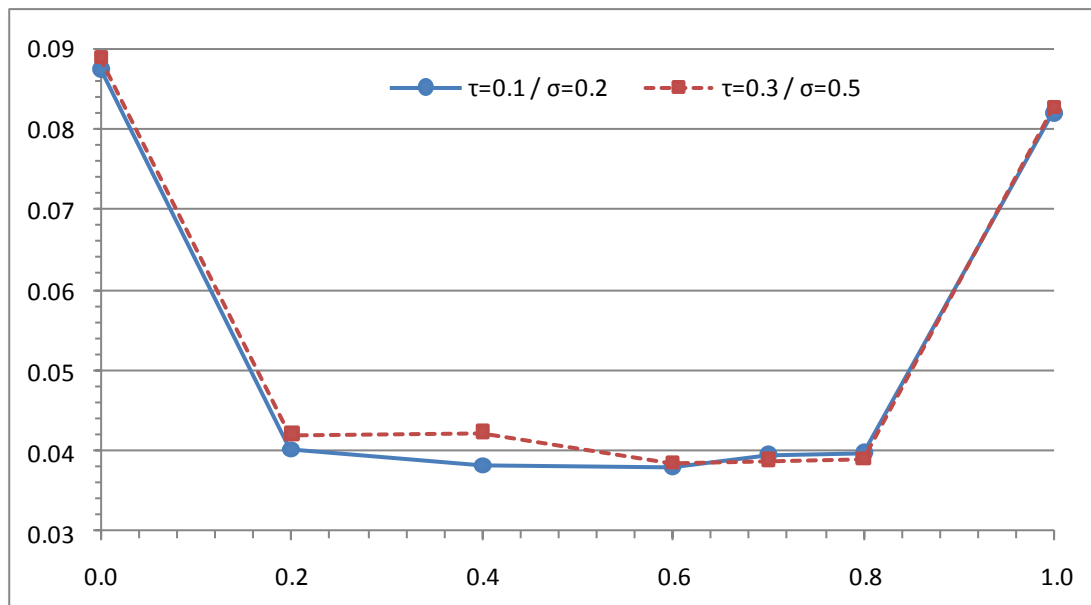


Figure 5.3 - Cooperation probability “p” (x-axis) tests with MAE values in the y-axis (lower value is better).

When looking at the results in Table 5.1 and Figure 5.3, one concludes that the cooperation probability parameter is very important to the evolution of the EPSO algorithm.

The extreme values of “p” (0 or 1) are the ones giving the worst results, as expected, because a very low value doesn’t take advantage of the swarm numbers cooperating to search for the optimum, and a very high value doesn’t allow for the particles to sweep the space more locally. Observing the graph in Figure 5.3, it’s obvious that the best values are in the interval [0.6-0.8]. When using Parzen window value $\sigma=0.2$ and learning rate $\tau=0.1$, one also obtains the best results when compared to the other ones ($\sigma=0.5$ and $\tau=0.3$).

The best values are obtained, for both tests, is when using a cooperation probability $p=0.6$, thus for the remaining tests of this chapter this is the value used.

5.2 Determining the Parzen window size “σ” and learning parameter “τ”

Just like the cooperation probability parameter, the learning parameter “τ” is also externally fixed. The purpose of this parameter, on EPSO, is to mutate the weights of each particle, thus creating a new position for it in the search space by changing how the terms inertia, memory, cooperation evolve when applied to the movement equations as described in Annex C. The weight mutation of normal Gaussian distribution, using the learning parameter “τ”, is given by:

$$W^k = W^{k-1} + [N(0,1) \times \tau] \quad (5.1)$$

Where:

W^k - New (mutated) weight;

W^{k-1} - Weight from previous iteration;

$N(0,1)$ - Normal Gaussian distribution;

τ - Learning parameter.

The learning parameter is thus as important as the cooperation parameter, since it affects the particle’s evolution by mutating its internal parameters in each iteration. This needs to be optimized too, so the algorithm can progress as efficiently as possible, avoiding premature convergence and also getting stuck on local optimums. This parameter is also what sets the EPSO apart from the PSO, because it introduces evolution programming/evolution strategies (EP/ES) resulting in competition between the parents and descendents through an elitist tournament¹². The higher the learning parameter, the most different is the descendent compared to the parent, but that doesn’t necessarily mean that it is a good thing. That’s why it is important to find the optimal value of the learning parameter.

Moreover, there is also the Parzen window size “σ” that greatly affects the resulting optimization of the MCC and evolution of the EPSO algorithm. Therefore, it is essential that these two parameters are compared so one can extrapolate which combination gives the best results. To that end, several tests were done with different combinations that are represented in Table 5.2. The test’s configuration is the same as the previous one, except for the variable parameters (τ and σ). The graphic in Figure 5.4 is the representation of the same test results as shown in Table 5.2.

Table 5.2 - Learning parameter “τ” and Parzen Window size “σ” tests (best value in gray). For reference, the mean absolute error (MAE) value for MSE is 0.090086.

		Learning parameter (τ)				
		0.1	0.3	0.5	0.7	0.9
Window Size (σ)	0.1	0.04077	0.04582	0.04646	0.04509	0.04105
	0.2	0.03752	0.04067	0.03948	0.04150	0.04404
	0.4	0.03784	0.03846	0.04086	0.04001	0.04346
	0.6	0.03817	0.03916	0.04135	0.03953	0.04169
	0.8	0.04494	0.04443	0.04496	0.04715	0.04599
	1.0	0.05327	0.05327	0.05152	0.05467	0.05424

¹² In elitist selection only the best particles survive to the next generation.

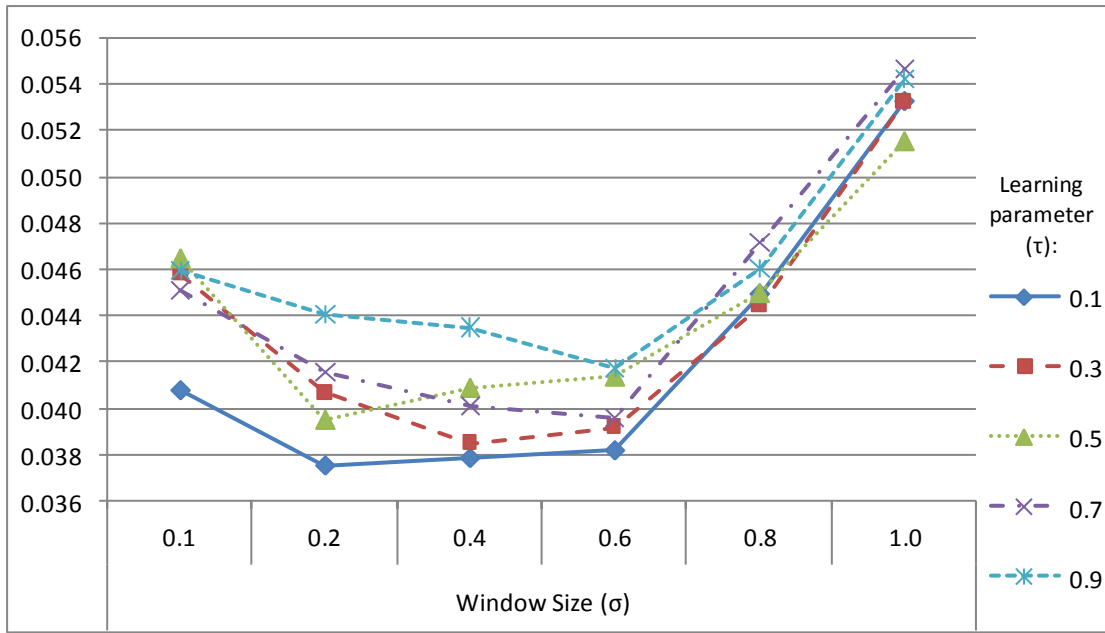


Figure 5.4 - Learning parameter “ τ ” and Parzen Window size “ σ ” tests (lower value is best). For reference, the mean absolute error (MAE) value for MSE is 0.090086. The y-axis represents the MAE.

By close inspection of the graph in Figure 5.4, it’s easy to notice how each combination of parameters affects the mean absolute error (MAE) of the maximum correntropy criterion (MCC). As a result, one observes that for Parzen window sizes (σ) between 0.1 and 0.6 the best results are given by the learning parameter (τ) equal to 0.1. For higher values of σ it is more effective to use the parameters $\tau=0.3$ for a $\sigma=0.8$ and $\tau=0.5$ for a $\sigma=1.0$.

These results can be explained by the inherent behaviour of the correntropy criterion function that has many local optimums and, depending on the Parzen window size, is spikier or smoother. The smaller the window size the spikier the function gets, thus creating a more difficult task for the EPSO convergence. It’s apparent that when using a narrow window size, the best results are given by the smaller learning parameters. It can be easily explained by the fact that, because of the problem dimension and fitness (optimization) function used in EPSO, if the mutations are big the algorithm has a difficult time progressing through, probably since some steep and narrow inclines can show up and the algorithm misses it.

This way, the EPSO can evolve much smoother and progressively on small learning parameters. Therefore, when using higher window sizes (0.6+) the difference is not that significant with different learning parameters, for the opposite reason as before (smother, less spiky and wider curves).

5.3 Numerical examples using AC 24-bus system

In this sub-chapter some examples will be done using different configurations and errors to prove that the MCC can also be successfully used in AC model power systems. It is important to emphasize that, since the optimization algorithm being used is the EPSO and, just like all EA's, needs to be executed several times for the results to be robust. Thus, 10 runs will be done for each test and the results will be averaged. Additionally, the best external EPSO parameters, elected in the previous sub-chapters, will be used in all examples unless otherwise mentioned. The errors mentioned before only affect the active power of the measurement. Once again the number of iterations used is **6000** with a population of **60** particles, resulting in **720000** fitness evaluations.

Essentially, the objective is to replicate the same results obtained in the DC system, on Chapter 4, to the 24-bus AC system.

Since the MSE calculations are done by iterative algorithm (Newton-Raphson) and MCC is done using evolutionary programming (EPSO), only the results and respective conclusions will be shown, given that the formulation of the NR algorithm is done in Chapter 3 and the EPSO doesn't have mathematical equations one can extract (black-box algorithm).

5.3.1 Example 1: Base case with error of 5% in line 8 (5-10)

This first example is just introducing a relatively small error to a line that connects node 5 to node 10. This error shouldn't affect the MSE state estimator results that much but the MCC should give better results comparatively.

A simple analysis of Figure 5.5 allows us to see how the MCC minimized all errors around zero except for the bad measurement P_{510} (line 5-10). Also, it's evident that the MSE criterion contaminated the other errors because of the effect of the bad measurement, especially the closest nodes' injected powers 5, 2 and 1. This once again proves how a bad measurement can negatively affect MSE's optimization.

Figure 5.6 excludes the bad measurement from the representation, thus giving a much better idea of the huge reduction in error's variance by MCC.

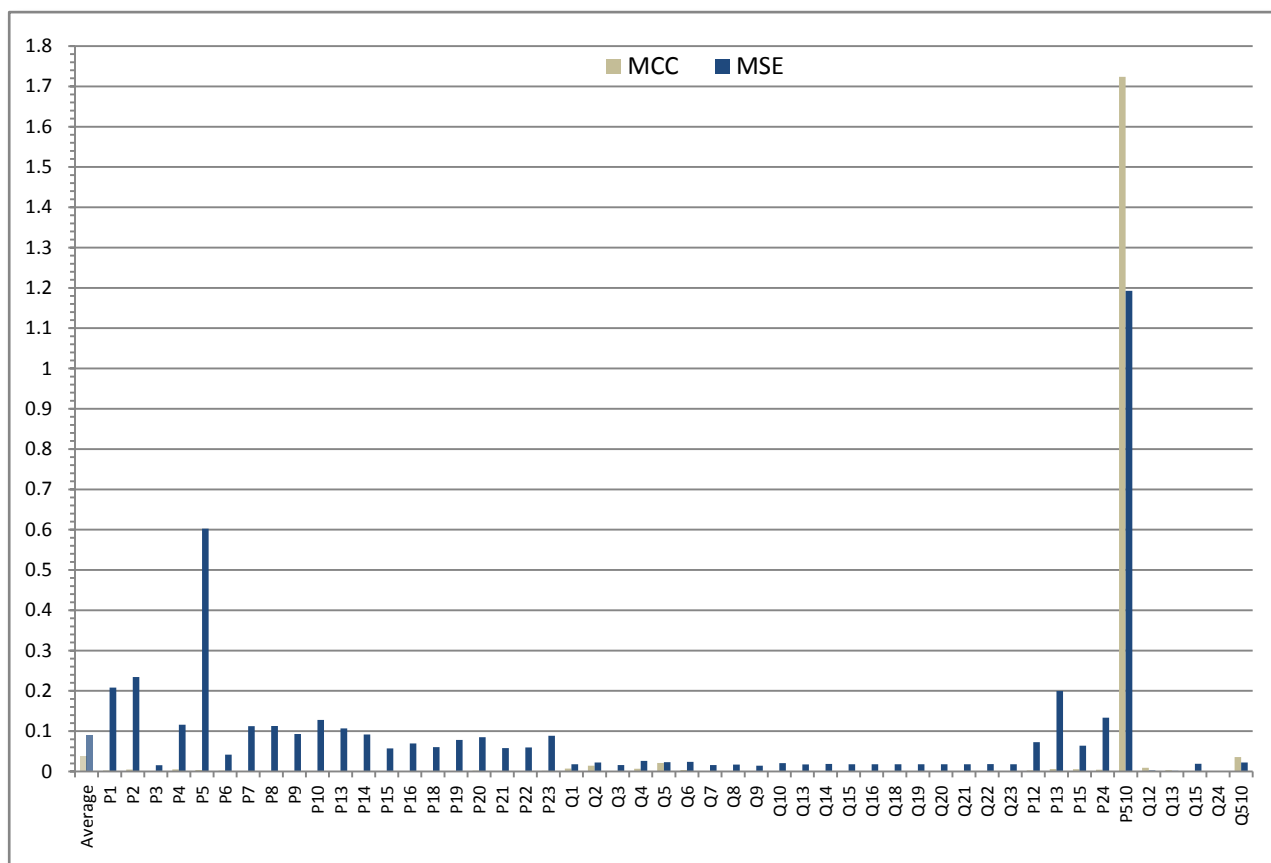


Figure 5.5 - Line 8 with +5% error: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE ("Average").

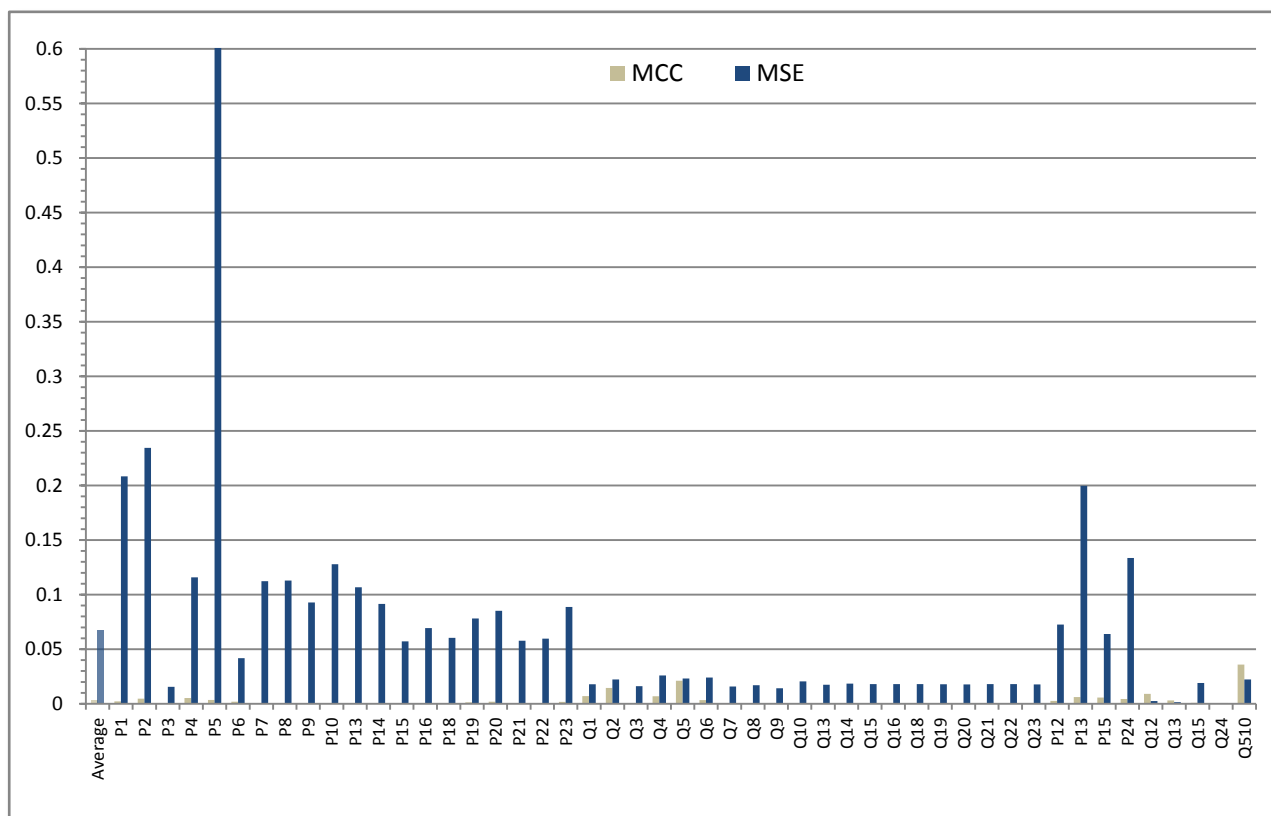


Figure 5.6 - Same graph as in Figure 5.5 excluding line 8 gross error. MAE recalculated.

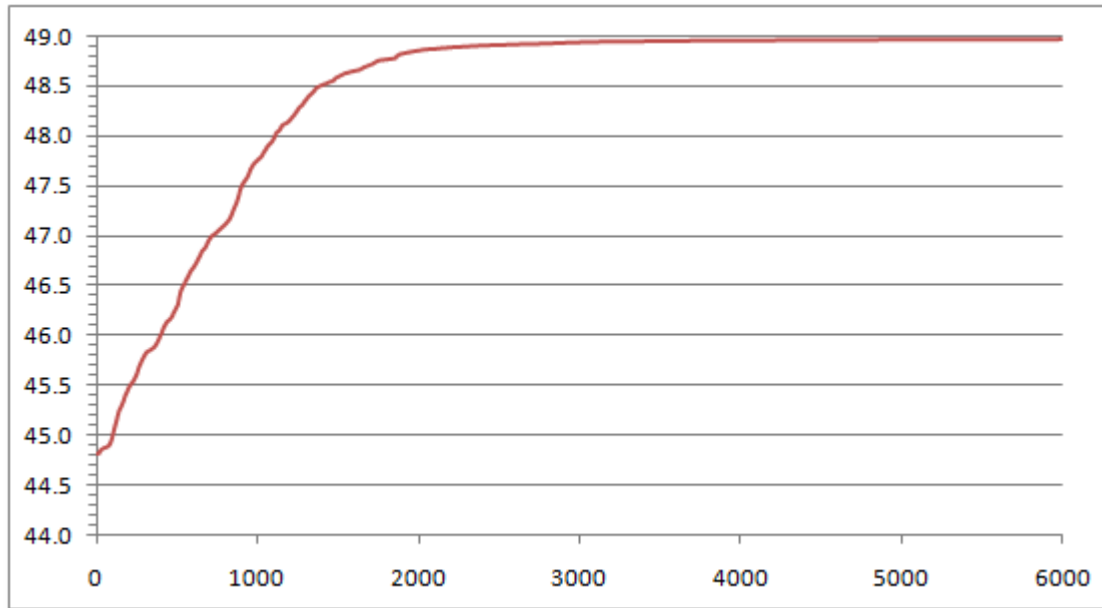


Figure 5.7 - EPSO fitness correntropy function evolution of example 1. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.

Figure 5.7 shows how the 10 run average of the EPSO evolves through the 6000 iterations. Note that the initial correntropy value is around 44.8. This is due to the Parzen window size used. If one would use a narrower window size the correntropy value would be smaller, and in the opposite case, a wider window size would give a higher correntropy value, as can be seen throughout this chapter. If the σ value is high enough MCC should, theoretically, have an approximate behaviour as MSE criterion, thus starting at a value much closer to the maximum correntropy value.

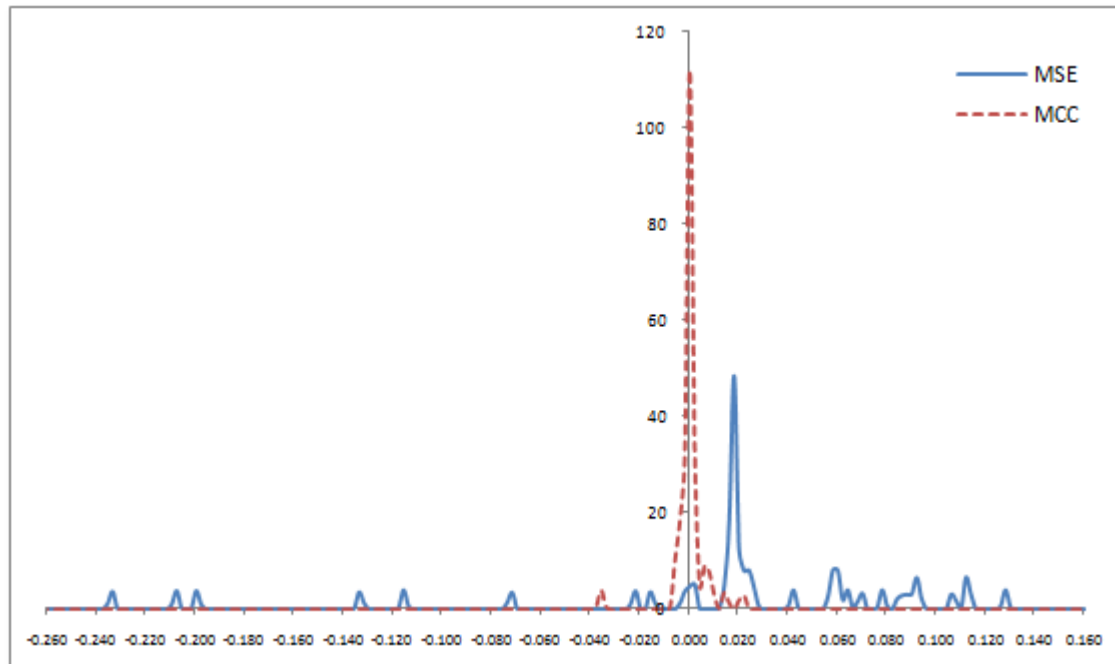


Figure 5.8 - Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.

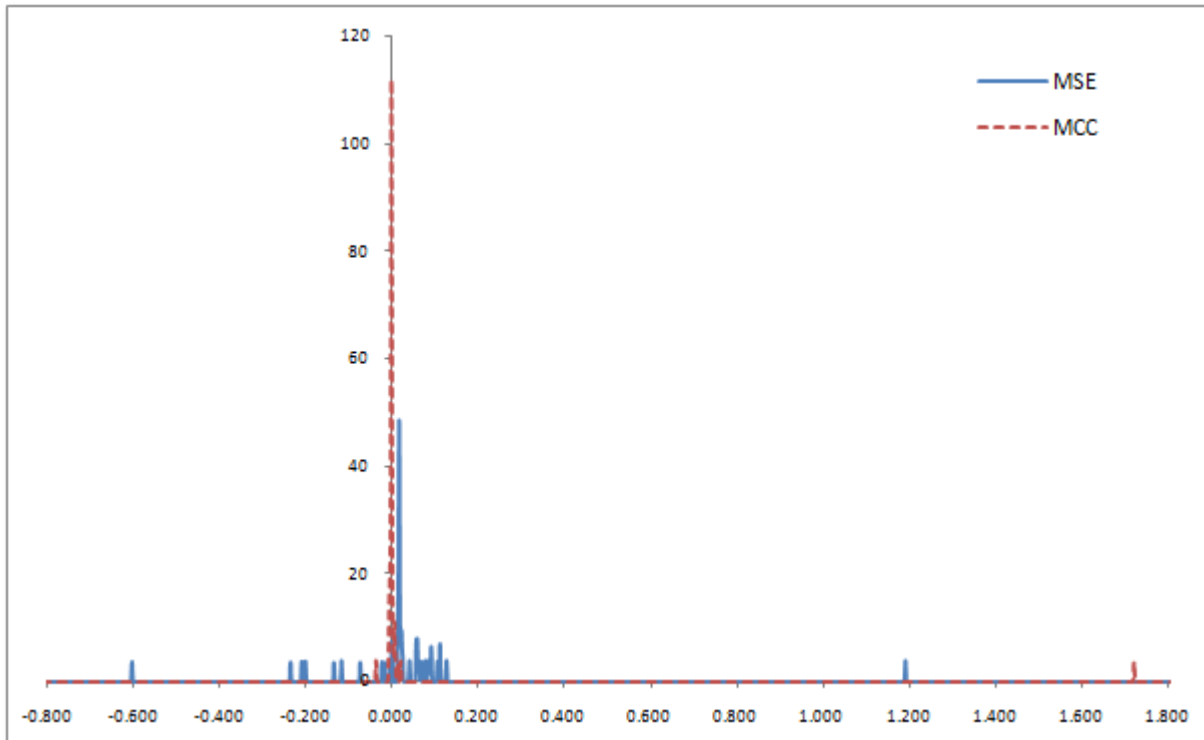


Figure 5.9 - Full Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.

In Figure 5.8 and Figure 5.9 there is a Gaussian distribution of the measurement errors, differing only in the inclusion of all the errors on the latter for a broader view, including the outlier isolated at around 1.7. On the other hand, Figure 5.8 shows only the errors much closer to zero where there is a big concentration of them, as this allows for a more precise analysis of the errors' distribution. For both these graphs, a very small $\sigma'=0.001$ is used.

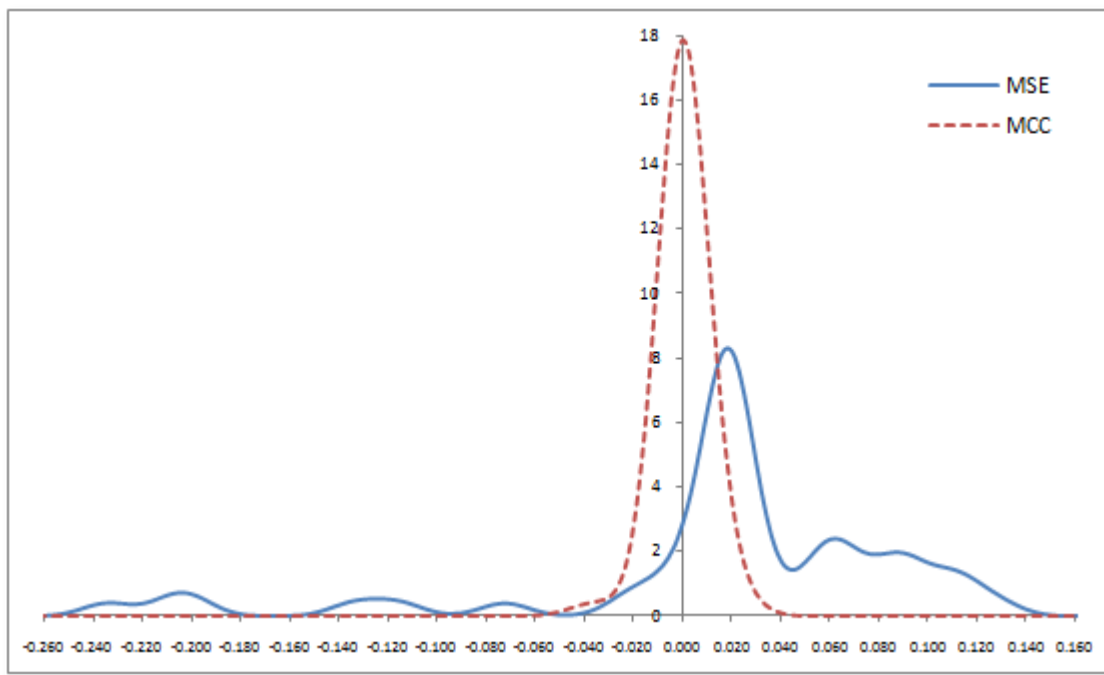


Figure 5.10 - Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.01$. The x-axis represents error values in MW/Mvar.

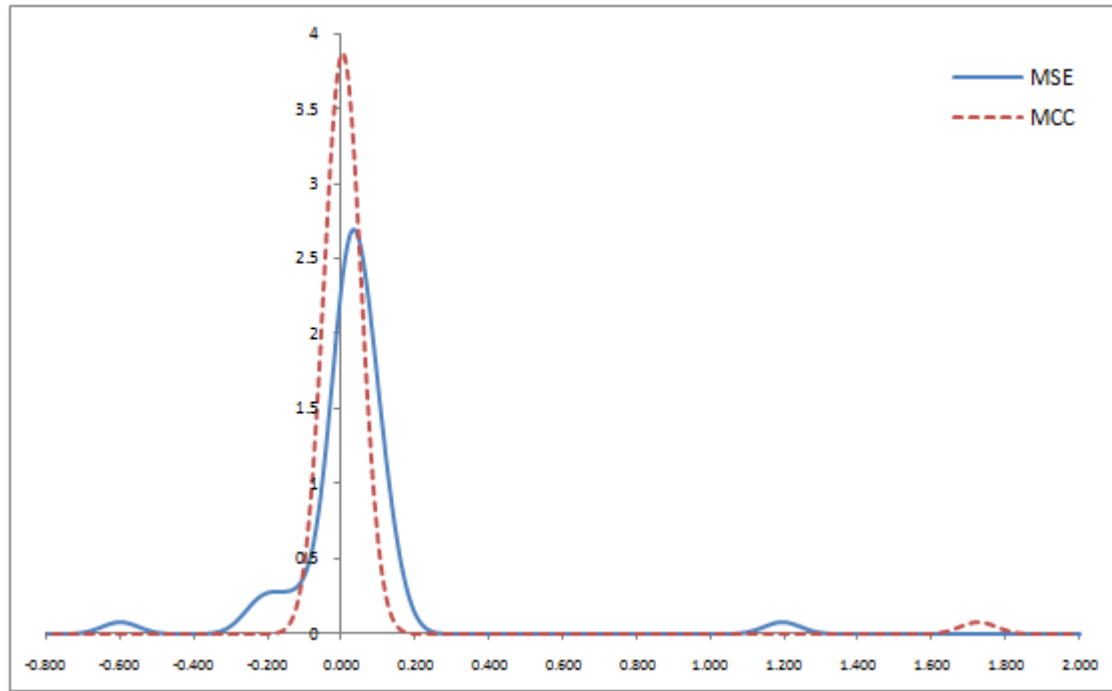


Figure 5.11 - Full Gaussian error distribution of example 1 using a Parzen window size $\sigma=0.2$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.

By using a small (narrow) Parzen window size $\sigma'=0.001$ as scaling parameter, it's easy to pinpoint the errors' location, thus one can see how the MSE criterion spreads the errors being contaminated by the introduced bad measurement (line 8 (5-10)). Once again, the MCC is able to find a much improved feasible solution to the state estimation problem by kicking the outlier away and compressing the remaining errors close to zero "ignoring" its effect.

In Figure 5.10 and Figure 5.11, a different Parzen window size scaling parameter is used; not to confuse with the MCC optimization Parzen window size, as that is still fixed in 0.2 for this example. The first figure uses a $\sigma'=0.01$ and, like Figure 5.8, focuses on the smaller errors close to zero. In this graph, the difference between the MSE criterion and the MCC error distribution is remarkable. The latter graph, like Figure 5.9, has a much wider x-axis range, covering all errors and allowing the observation of the outlier like previously mentioned. It should be noted that there was a need to increase the σ' scale parameter value to 0.05 in the graph to allow for a better understanding of the curves.

Finally, in Table 5.3, the values of the initial value for the measurements used by minimum square error (MSE or WLS) criterion and the maximum correntropy criterion (MCC) in the respective RTSE and EPSO programs are shown. The table also presents the seed and changed measurements and respective calculated values. Furthermore, if one analyses the MCC vs. MSE differences it's evident that the first has much better results than the latter, by being able to minimize almost all the errors being seemingly unaffected by the gross error. Also, when comparing the same error difference, the gross error increases in the MCC getting very close to the "real" value in the seed values. Three of the errors besides the gross are slightly worse in value than with the MSE criterion. This could be due to the slight fluctuation in solutions of the EPSO causing it to converge to a less than ideal solution. However, the deviation is very small and doesn't affect the outcome of the main results.

Table 5.3 - Seed, initial measurement errors and estimated values for example 1.

	Measurements			Errors		Estimated Values		Differences	
	ID	Seed	Error added	MSE	MCC	MSE	MCC	MCC vs MSE	MCC vs Seed
Injected Powers	P1	150.000000	150.000000	2.08E-01	2.28E-03	149.791687	149.997716	-0.206029	-2.284E-03
	P2	10.000000	10.000000	2.34E-01	4.70E-03	9.765585	9.995297	-0.229712	-4.703E-03
	P3	-44.205356	-44.205356	1.55E-02	1.69E-04	-44.220892	-44.205525	-0.015367	1.690E-04
	P4	-18.255175	-18.255175	1.16E-01	5.34E-03	-18.371073	-18.260519	-0.110553	5.344E-03
	P5	-17.436557	-17.436557	6.02E-01	3.52E-03	-18.038927	-17.440080	-0.598846	3.523E-03
	P6	-33.399602	-33.399602	4.18E-02	1.95E-03	-33.441430	-33.401556	-0.039874	1.954E-03
	P7	10.000000	10.000000	1.12E-01	5.16E-04	9.887700	9.999484	-0.111785	-5.157E-04
	P8	-41.995088	-41.995088	1.13E-01	3.83E-04	-42.107932	-41.995472	-0.112460	3.834E-04
	P9	-42.977430	-42.977430	9.28E-02	3.84E-04	-43.070225	-42.977814	-0.092411	3.844E-04
	P10	-47.889136	-47.889136	1.28E-01	1.54E-04	-48.017053	-47.889290	-0.127763	1.543E-04
	P13	10.000000	10.000000	1.07E-01	4.59E-04	9.893294	9.999541	-0.106246	-4.595E-04
	P14	-47.643550	-47.643550	9.16E-02	6.78E-04	-47.735106	-47.644228	-0.090878	6.775E-04
	P15	10.000000	10.000000	5.73E-02	4.73E-05	9.942745	9.999953	-0.057208	-4.726E-05
	P16	10.000000	10.000000	6.94E-02	2.03E-04	9.930628	9.999797	-0.069169	-2.030E-04
	P18	10.000000	10.000000	6.04E-02	1.84E-04	9.939648	9.999816	-0.060168	-1.844E-04
	P19	-44.450941	-44.450941	7.81E-02	1.28E-03	-44.529055	-44.452225	-0.076830	1.283E-03
	P20	-31.434920	-31.434920	8.51E-02	1.92E-03	-31.520059	-31.436843	-0.083215	1.924E-03
	P21	10.000000	10.000000	5.78E-02	1.52E-04	9.942203	9.999848	-0.057644	-1.524E-04
	P22	10.000000	10.000000	5.97E-02	1.09E-04	9.940268	9.999891	-0.059623	-1.093E-04
	P23	143.534061	143.534061	8.88E-02	1.73E-03	143.445293	143.532328	-0.087035	-1.733E-03
	Q1	-11.491917	-11.491917	1.80E-02	7.02E-03	-11.509872	-11.498939	-0.010932	7.022E-03
	Q2	16.142114	16.142114	2.22E-02	1.45E-02	16.119870	16.127609	-0.007739	-1.450E-02
	Q3	-14.529598	-14.529598	1.61E-02	3.83E-04	-14.545675	-14.529981	-0.015694	3.830E-04
	Q4	-6.000186	-6.000186	2.60E-02	6.92E-03	-6.026201	-6.007105	-0.019096	6.919E-03
	Q5	-5.731119	-5.731119	2.31E-02	2.12E-02	-5.754237	-5.752319	-0.001919	2.120E-02
	Q6	-10.977918	-10.977918	2.40E-02	3.40E-03	-11.001944	-10.981316	-0.020628	3.398E-03
	Q7	20.800598	20.800598	1.59E-02	3.17E-04	20.784735	20.800281	-0.015546	-3.174E-04
	Q8	-13.803118	-13.803118	1.70E-02	3.81E-05	-13.820115	-13.803156	-0.016959	3.808E-05
	Q9	-14.125998	-14.125998	1.43E-02	7.27E-04	-14.140282	-14.126725	-0.013557	7.268E-04
	Q10	-15.740398	-15.740398	2.06E-02	3.75E-04	-15.760951	-15.740773	-0.020178	3.751E-04
	Q13	46.060811	46.060811	1.74E-02	1.51E-04	46.043396	46.060660	-0.017264	-1.509E-04
	Q14	-15.659678	-15.659678	1.85E-02	1.51E-04	-15.678191	-15.659828	-0.018363	1.505E-04
	Q15	16.787847	16.787847	1.80E-02	1.77E-05	16.769804	16.787829	-0.018025	-1.771E-05
	Q16	38.978973	38.978973	1.80E-02	1.85E-04	38.960932	38.978788	-0.017857	-1.848E-04
	Q18	-1.262795	-1.262795	1.81E-02	1.32E-04	-1.280853	-1.262927	-0.017926	1.315E-04
	Q19	-14.610318	-14.610318	1.79E-02	5.13E-04	-14.628260	-14.610831	-0.017428	5.135E-04
	Q20	-10.332159	-10.332159	1.78E-02	4.92E-04	-10.349946	-10.332650	-0.017296	4.916E-04
	Q21	-1.492295	-1.492295	1.80E-02	2.79E-04	-1.510342	-1.492574	-0.017768	2.788E-04
	Q22	-1.279603	-1.279603	1.81E-02	3.40E-04	-1.297685	-1.279943	-0.017743	3.398E-04
	Q23	15.323757	15.323757	1.77E-02	4.16E-04	15.306051	15.323341	-0.017290	-4.165E-04
Line Power flows	P12	62.912595	62.912595	7.25E-02	2.65E-03	62.840050	62.909941	-0.069891	-2.654E-03
	P13	34.079340	34.079340	2.00E-01	6.14E-03	33.879635	34.073197	-0.193562	-6.143E-03
	P15	53.008065	53.008065	6.39E-02	5.77E-03	52.944128	53.002292	-0.058164	-5.773E-03
	P24	37.936167	37.936167	1.34E-01	4.35E-03	37.802644	37.931814	-0.129170	-4.353E-03
	P510	34.958731	36.706668	1.19E+00	1.72E+00	35.513780	34.982992	0.530788	2.426E-02
	Q12	-11.496704	-11.496704	2.45E-03	9.13E-03	-11.499157	-11.505835	0.006678	9.131E-03
	Q13	-1.017736	-1.017736	1.41E-03	3.08E-03	-1.019151	-1.020813	0.001662	3.077E-03
	Q15	1.022523	1.022523	1.90E-02	9.69E-04	1.003531	1.021554	-0.018023	-9.690E-04
	Q24	1.951852	1.951852	7.39E-04	4.25E-04	1.951113	1.951427	-0.000314	-4.246E-04
	Q510	-7.081546	-7.081546	2.24E-02	3.59E-02	-7.103900	-7.117440	0.013540	3.589E-02

5.3.2 Example 2: Base case with error of 20% in line 8 (5-10)

In this example, the measurement error P_{510} (line 8) introduced in example 1 is increased to 20%, thus creating a far worse case scenario and it's expected that the MSE criterion will give a worse solution than in example 1, further contaminating the remaining errors. On the other hand, MCC should give similar results as in example 1, this way confirming its L0 metric behaviour over the gross errors.

For the EPSO to converge to a feasible solution there was a need to change the Parzen window size to **0.4**. It's easy to understand why this was needed; since the MSE errors all increased in magnitude, one had to widen the Parzen window size in order to include the “small” errors in the optimization while still isolating the gross error as outlier.

Comparing the results between Figure 5.5 and Figure 5.12, it's clear that all MSE's errors increased in magnitude, confirming the gross error contaminating the other errors in an attempt, by the MSE algorithm, to minimize all errors' variance. While, in example 1, the gross error had a 1.7 deviation value, now it has close to 7! Moreover, the MCC's results, as expected, are very similar to the previous example, confirming its L2 and L1 metric properties on the small errors. Like in the previous example, Figure 5.13 represents the same results as Figure 5.12, without using the gross measurement in the graphic data, giving once again a sense of how dramatic the reduction made by MCC is.

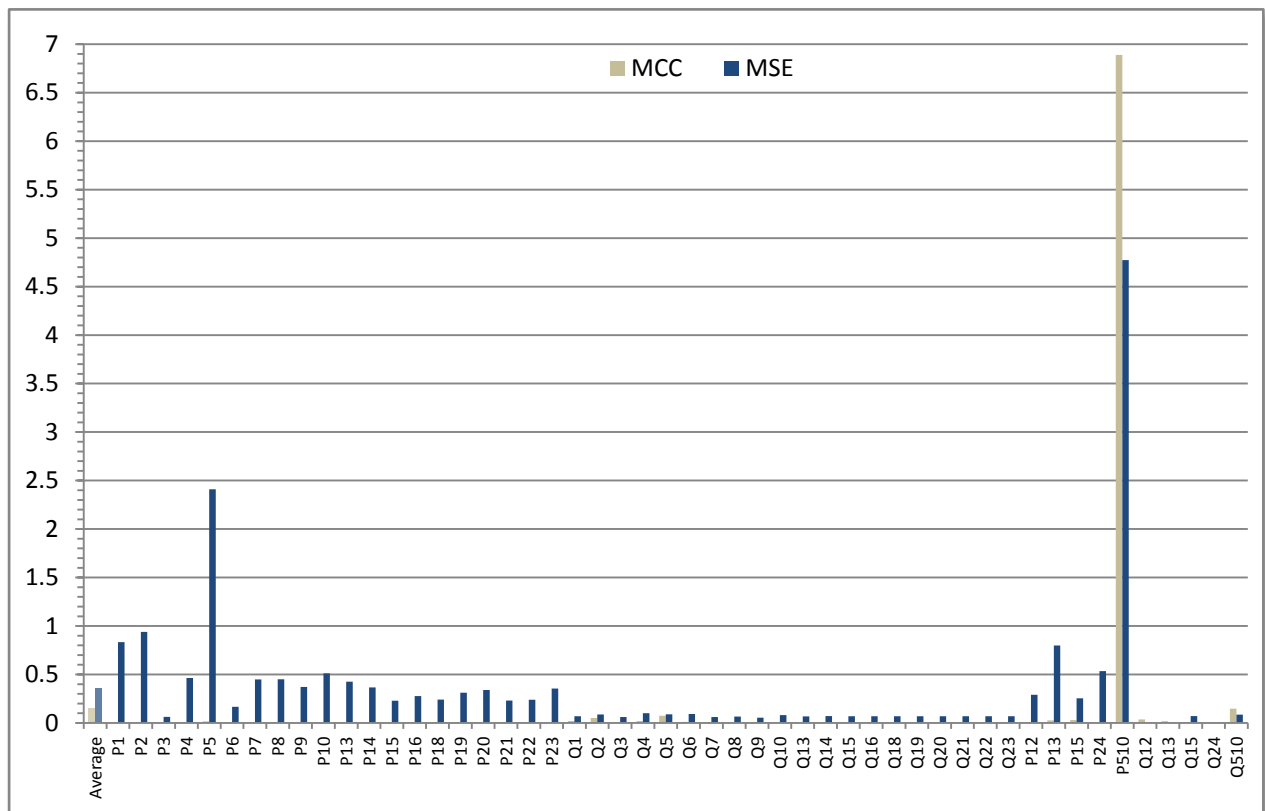


Figure 5.12 - Line 8 with +20% error: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE (“Average”).

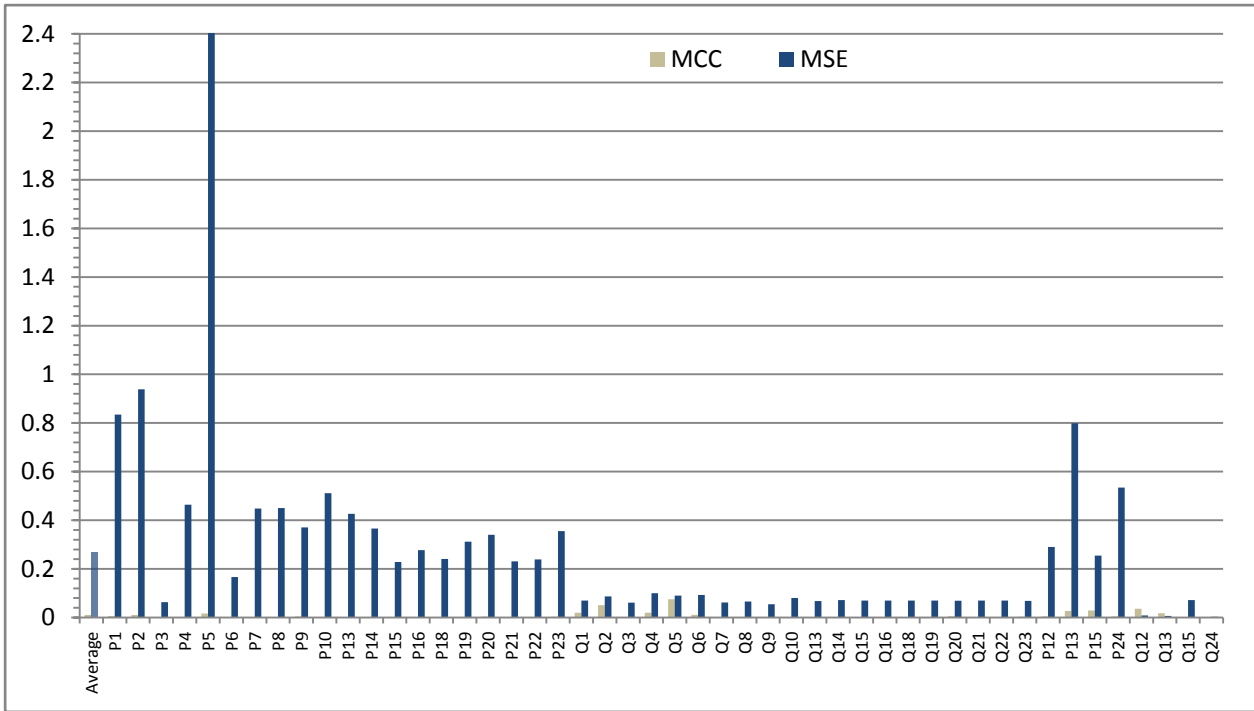


Figure 5.13 - Same graph as in Figure 5.12 excluding line 8 gross error. MAE recalculated.

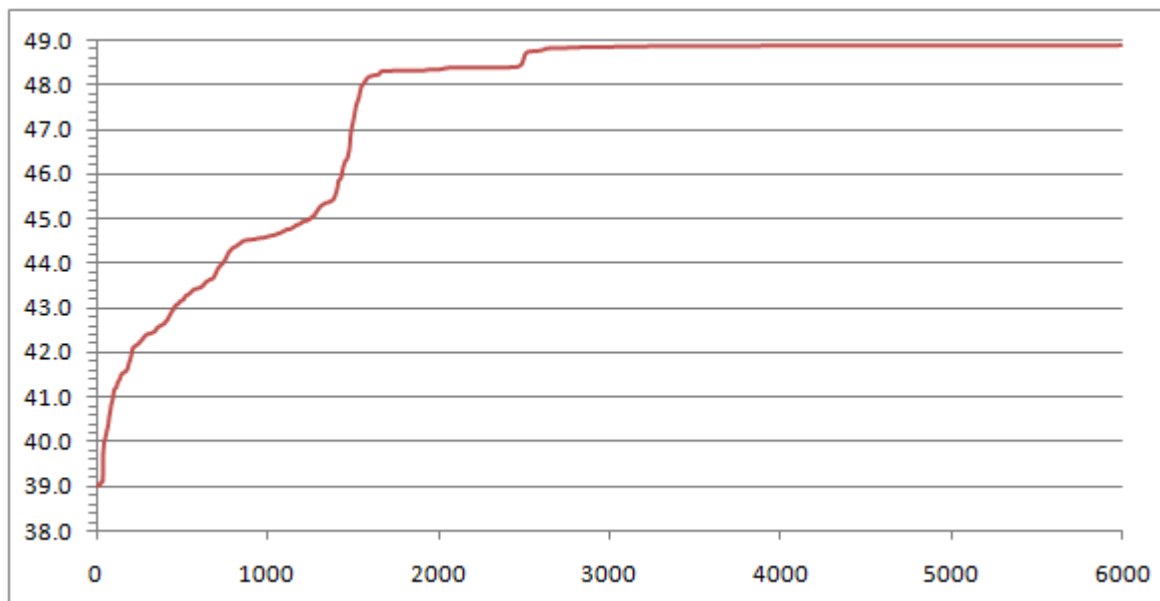


Figure 5.14 - EPSO fitness correntropy function evolution of example 2. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.

Again, Figure 5.14 shows how the 10 run average of the EPSO evolves through the 6000 iterations. Note that, this time, the initial correntropy value is around 39. This is due to the used Parzen window size. The narrower the window size, the smaller the starting correntropy value and the wider the window, size the higher the starting correntropy value.

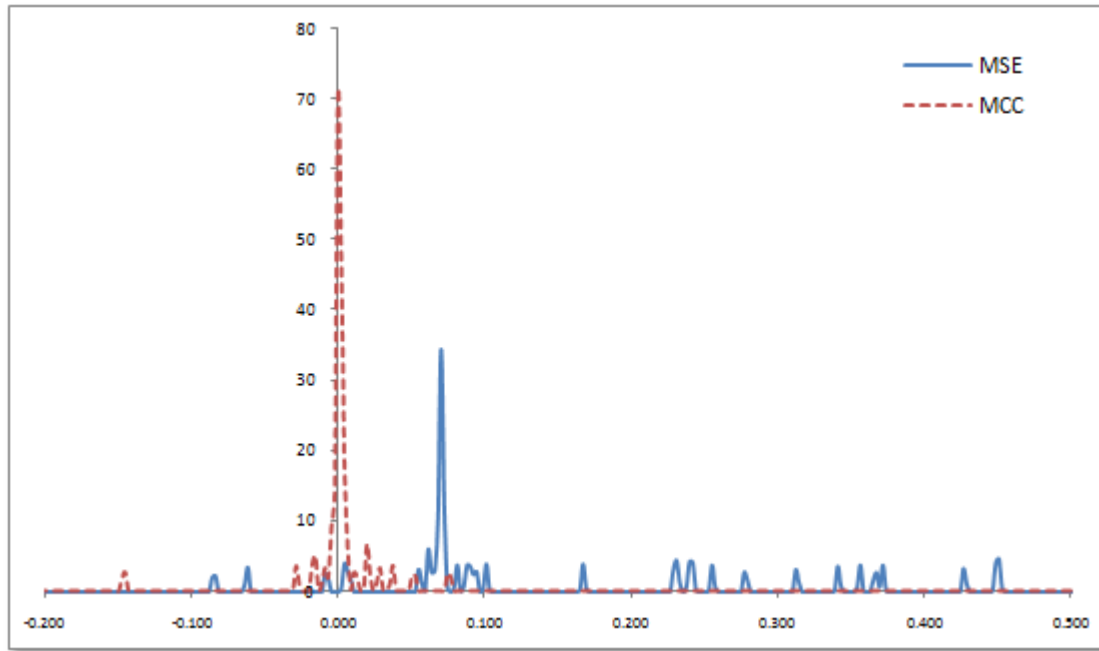


Figure 5.15 - Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.

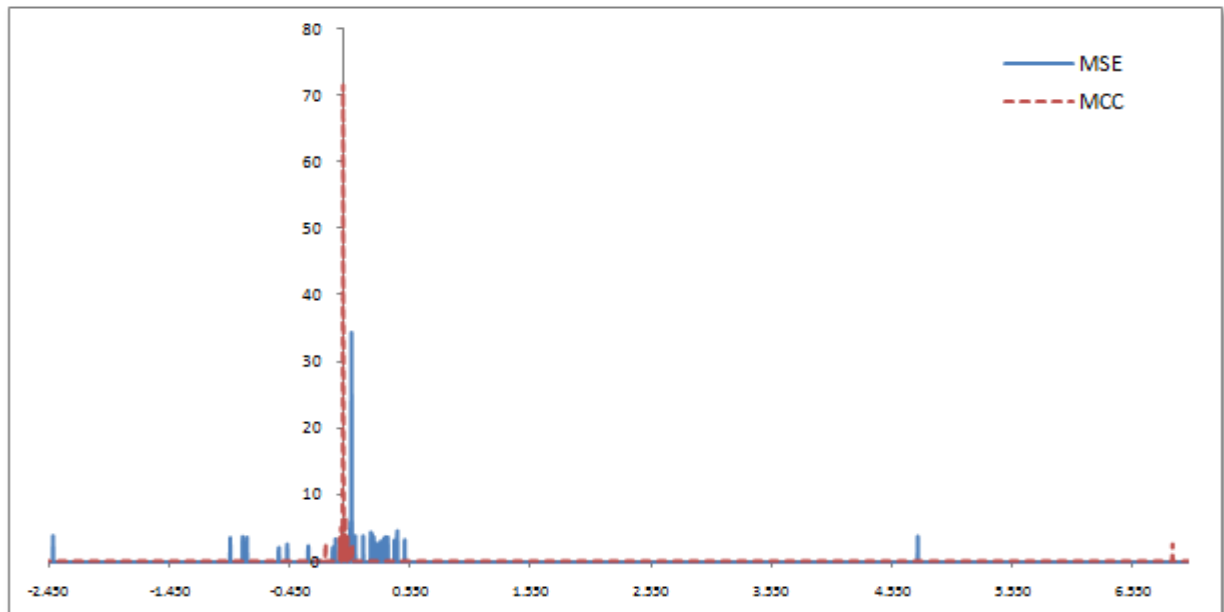


Figure 5.16 - Full Gaussian error distribution of example 2 u using a Parzen window size $\sigma=0.4$ and $\sigma'=0.001$. The x-axis represents error values in MW/Mvar.

Figure 5.15 and Figure 5.16 again represent the Gaussian distribution of the measurement errors, differing only in the inclusion of all the errors on the latter for a broader view, including the outlier isolated at around 7.0. For both graphics, a very small scaling parameter $\sigma'=0.001$ is used, as it allows for a broad view of all error's location.

Comparing to the figures in example 1, it's obvious that the error distribution is now much more affected by the gross measurement error (P_{510}). However, the MCC gives almost the same results as in example 1, again confirming the L0 metric already mentioned. This is represented in Table 5.5, where the MCC results from example 1 and 2 are compared against each other and the seed values.

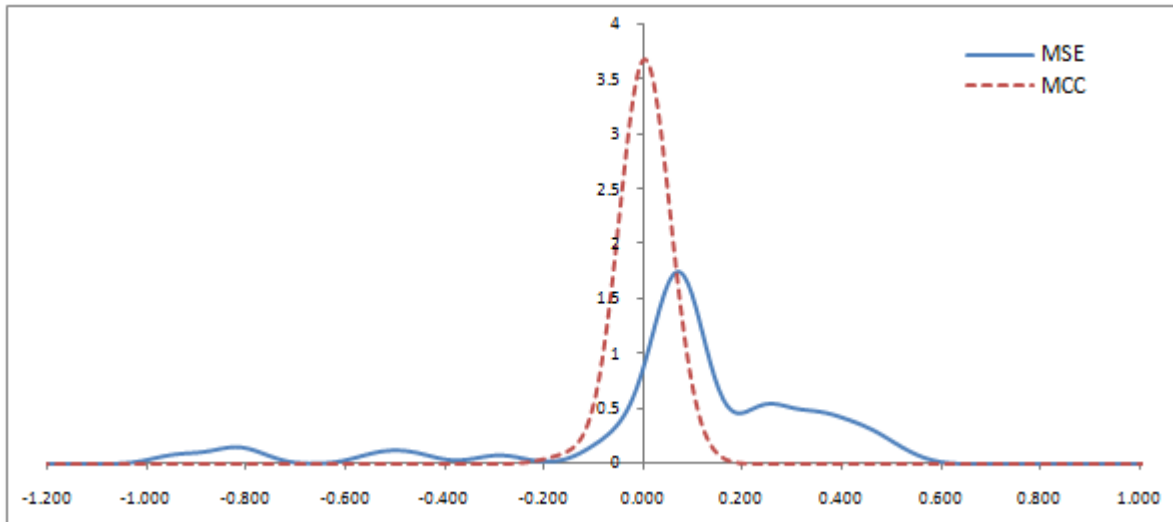


Figure 5.17 - Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.

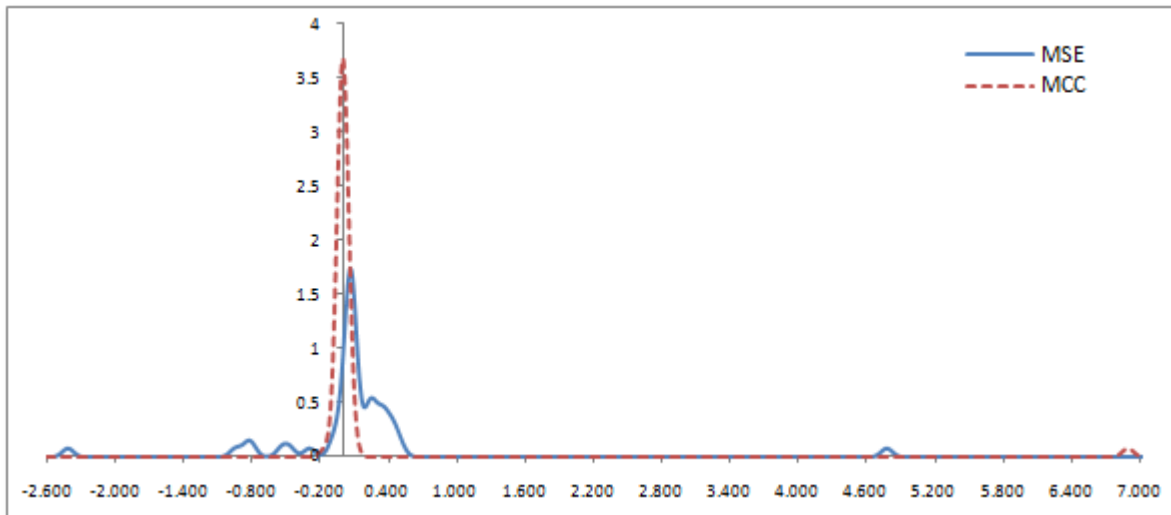


Figure 5.18 - Full Gaussian error distribution of example 2 using a Parzen window size $\sigma=0.4$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.

Again, by increasing the Parzen window size scaling parameter (not to confuse with the MCC optimization Parzen window) one gets a smoother view of the error distributions. It is obvious that MSE curve is nothing close to a Gaussian distribution and that is the cause of the poor behaviour of its optimization capacity.

Lastly, in Table 5.4, the values of the seed, starting measurement vector used by the MSE criterion and the MCC, and the respective errors are presented. Comparing the MSE vs. MCC values in the table, it's obvious the huge advance in the minimization of all the MCC errors except for the outlier that sees its error increased. The most curious thing to see in the table is the comparison between the MCC result and the seed. If one analyses the introduced gross error's values (P_{510}) it's clear that the MCC "tried" to correct the gross error to the most coherent result possible, thus the estimated MCC P_{510} value being very close to the seed value. This is a very important result, because it shows the great advantage of an optimization algorithm such as the MCC over the traditional MSE criterion in state estimation.

Table 5.4 - Seed, initial measurement errors and estimated values for example 2.

	Measurements			Errors		Estimated Values		Differences	
	ID	Seed	Error added	MSE	MCC	MSE	MCC	MCC vs MSE	MCC vs Seed
Injected Powers	P1	150.000000	150.000000	-8.34E-01	-6.08E-03	150.834353	150.006080	-0.828274	6.080E-03
	P2	10.000000	10.000000	-9.38E-01	-1.02E-02	10.938381	10.010236	-0.928145	1.024E-02
	P3	-44.205356	-44.205356	-6.29E-02	1.65E-03	-44.142450	-44.207011	-0.061252	1.655E-03
	P4	-18.255175	-18.255175	-4.64E-01	-1.90E-03	-17.791053	-18.253273	-0.462220	-1.902E-03
	P5	-17.436557	-17.436557	-2.41E+00	-1.68E-02	-15.026773	-17.419792	-2.393019	-1.677E-02
	P6	-33.399602	-33.399602	1.67E-01	-1.18E-03	-33.566127	-33.398423	-0.165345	-1.179E-03
	P7	10.000000	10.000000	4.48E-01	-6.68E-06	9.551674	10.000007	-0.448319	6.680E-06
	P8	-41.995088	-41.995088	4.50E-01	-8.94E-04	-42.445341	-41.994194	-0.449358	-8.945E-04
	P9	-42.977430	-42.977430	3.71E-01	-4.84E-03	-43.348121	-42.972590	-0.365852	-4.839E-03
	P10	-47.889136	-47.889136	5.11E-01	-5.93E-04	-48.400365	-47.888543	-0.510637	-5.931E-04
	P13	10.000000	10.000000	4.27E-01	1.30E-03	9.573392	9.998705	-0.425313	-1.295E-03
	P14	-47.643550	-47.643550	3.66E-01	1.67E-03	-48.009279	-47.645224	-0.364055	1.674E-03
	P15	10.000000	10.000000	2.29E-01	-4.92E-04	9.771268	10.000492	-0.228240	4.919E-04
	P16	10.000000	10.000000	2.77E-01	1.14E-04	9.722761	9.999886	-0.277126	-1.137E-04
	P18	10.000000	10.000000	2.41E-01	-8.39E-04	9.758835	10.000839	-0.240326	8.389E-04
	P19	-44.450941	-44.450941	3.12E-01	2.13E-03	-44.763226	-44.453071	-0.310155	2.129E-03
	P20	-31.434920	-31.434920	3.41E-01	4.00E-03	-31.775538	-31.438915	-0.336622	3.995E-03
	P21	10.000000	10.000000	2.31E-01	-7.32E-04	9.769088	10.000732	-0.230180	7.322E-04
	P22	10.000000	10.000000	2.39E-01	-5.00E-04	9.761241	10.000500	-0.238258	5.003E-04
	P23	143.534061	143.534061	3.55E-01	3.09E-03	143.178738	143.530970	-0.352232	-3.091E-03
	Q1	-11.491917	-11.491917	6.95E-02	1.95E-02	-11.561451	-11.511436	-0.050015	1.952E-02
	Q2	16.142114	16.142114	8.68E-02	5.09E-02	16.055305	16.091226	-0.035921	-5.089E-02
	Q3	-14.529598	-14.529598	6.10E-02	1.34E-03	-14.590638	-14.530936	-0.059702	1.338E-03
	Q4	-6.000186	-6.000186	1.00E-01	1.96E-02	-6.100501	-6.019832	-0.080669	1.965E-02
	Q5	-5.731119	-5.731119	8.99E-02	7.50E-02	-5.821031	-5.806094	-0.014937	7.497E-02
	Q6	-10.977918	-10.977918	9.30E-02	1.11E-02	-11.070909	-10.989044	-0.081865	1.113E-02
	Q7	20.800598	20.800598	6.19E-02	2.55E-04	20.738713	20.800343	-0.061630	-2.552E-04
	Q8	-13.803118	-13.803118	6.60E-02	-4.64E-04	-13.869161	-13.802654	-0.065578	-4.642E-04
	Q9	-14.125998	-14.125998	5.49E-02	1.10E-03	-14.180855	-14.127100	-0.053755	1.102E-03
	Q10	-15.740398	-15.740398	8.03E-02	-1.61E-03	-15.820716	-15.738785	-0.078705	-1.613E-03
	Q13	46.060811	46.060811	6.77E-02	-1.67E-04	45.993104	46.060978	-0.067540	1.668E-04
	Q14	-15.659678	-15.659678	7.18E-02	3.61E-04	-15.731437	-15.660039	-0.071398	3.608E-04
	Q15	16.787847	16.787847	6.98E-02	-1.59E-03	16.718029	16.789432	-0.068233	1.585E-03
	Q16	38.978973	38.978973	6.99E-02	-1.12E-03	38.909097	38.980091	-0.068759	1.118E-03
	Q18	-1.262795	-1.262795	6.99E-02	-6.42E-04	-1.332684	-1.262153	-0.069247	-6.418E-04
	Q19	-14.610318	-14.610318	6.95E-02	8.22E-04	-14.679819	-14.611140	-0.068679	8.216E-04
	Q20	-10.332159	-10.332159	6.89E-02	5.52E-03	-10.401097	-10.337675	-0.063422	5.517E-03
	Q21	-1.492295	-1.492295	6.98E-02	3.47E-04	-1.562130	-1.492642	-0.069488	3.469E-04
	Q22	-1.279603	-1.279603	7.00E-02	-1.73E-04	-1.349583	-1.279430	-0.069807	-1.728E-04
	Q23	15.323757	15.323757	6.86E-02	3.34E-03	15.255110	15.320421	-0.065312	-3.336E-03
Line Power flows	P12	62.912595	62.912595	-2.90E-01	-4.33E-03	63.202583	62.916930	-0.285654	4.335E-03
	P13	34.079340	34.079340	-7.99E-01	2.72E-02	34.878403	34.052169	-0.771893	-2.717E-02
	P15	53.008065	53.008065	2.55E-01	-2.89E-02	52.753367	53.036981	-0.225782	2.892E-02
	P24	37.936167	37.936167	-5.34E-01	4.30E-03	38.470482	37.931863	-0.530011	-4.304E-03
	P510	34.958731	41.950477	4.77E+00	6.89E+00	37.177447	35.061831	2.115617	1.031E-01
	Q12	-11.496704	-11.496704	-8.66E-03	3.62E-02	-11.488049	-11.532855	0.027496	3.615E-02
	Q13	-1.017736	-1.017736	6.68E-03	-1.76E-02	-1.024421	-1.000178	0.010874	-1.756E-02
	Q15	1.022523	1.022523	7.15E-02	9.27E-04	0.951019	1.021596	-0.070578	-9.267E-04
	Q24	1.951852	1.951852	3.51E-03	-2.11E-03	1.948340	1.953964	-0.001400	2.112E-03
	Q510	-7.081546	-7.081546	-8.55E-02	-1.47E-01	-6.996082	-6.934647	0.061435	-1.469E-01

Table 5.5 - Comparison between MCC results for example 1 and 2.

Measurements		MCC Results		Comparison
ID	Seed	Example 1	Example 2	Ex. 1 vs Ex. 2
P1	150.000000	149.997716	150.006080	8.3640E-03
P2	10.000000	9.995297	10.010236	1.4940E-02
P3	-44.205356	-44.205525	-44.207011	1.4856E-03
P4	-18.255175	-18.260519	-18.253273	-7.2467E-03
P5	-17.436557	-17.440080	-17.419792	-2.0289E-02
P6	-33.399602	-33.401556	-33.398423	-3.1330E-03
P7	10.000000	9.999484	10.000007	5.2234E-04
P8	-41.995088	-41.995472	-41.994194	-1.2778E-03
P9	-42.977430	-42.977814	-42.972590	-5.2234E-03
P10	-47.889136	-47.889290	-47.888543	-7.4739E-04
P13	10.000000	9.999541	9.998705	-8.3553E-04
P14	-47.643550	-47.644228	-47.645224	9.9632E-04
P15	10.000000	9.999953	10.000492	5.3917E-04
P16	10.000000	9.999797	9.999886	8.9286E-05
P18	10.000000	9.999816	10.000839	1.0233E-03
P19	-44.450941	-44.452225	-44.453071	8.4583E-04
P20	-31.434920	-31.436843	-31.438915	2.0719E-03
P21	10.000000	9.999848	10.000732	8.8462E-04
P22	10.000000	9.999891	10.000500	6.0963E-04
P23	143.534061	143.532328	143.530970	-1.3579E-03
Q1	-11.491917	-11.498939	-11.511436	1.2497E-02
Q2	16.142114	16.127609	16.091226	-3.6383E-02
Q3	-14.529598	-14.529981	-14.530936	9.5514E-04
Q4	-6.000186	-6.007105	-6.019832	1.2727E-02
Q5	-5.731119	-5.752319	-5.806094	5.3776E-02
Q6	-10.977918	-10.981316	-10.989044	7.7278E-03
Q7	20.800598	20.800281	20.800343	6.2239E-05
Q8	-13.803118	-13.803156	-13.802654	-5.0230E-04
Q9	-14.125998	-14.126725	-14.127100	3.7567E-04
Q10	-15.740398	-15.740773	-15.738785	-1.9880E-03
Q13	46.060811	46.060660	46.060978	3.1771E-04
Q14	-15.659678	-15.659828	-15.660039	2.1031E-04
Q15	16.787847	16.787829	16.789432	1.6031E-03
Q16	38.978973	38.978788	38.980091	1.3025E-03
Q18	-1.262795	-1.262927	-1.262153	-7.7334E-04
Q19	-14.610318	-14.610831	-14.611140	3.0814E-04
Q20	-10.332159	-10.332650	-10.337675	5.0252E-03
Q21	-1.492295	-1.492574	-1.492642	6.8116E-05
Q22	-1.279603	-1.279943	-1.279430	-5.1265E-04
Q23	15.323757	15.323341	15.320421	-2.9193E-03
P12	62.912595	62.909941	62.916930	6.9888E-03
P13	34.079340	34.073197	34.052169	-2.1028E-02
P15	53.008065	53.002292	53.036981	3.4689E-02
P24	37.936167	37.931814	37.931863	4.9260E-05
P510	34.958731	34.982992	35.061831	7.8839E-02
Q12	-11.496704	-11.505835	-11.532855	2.7020E-02
Q13	-1.017736	-1.020813	-1.000178	-2.0636E-02
Q15	1.022523	1.021554	1.021596	4.2222E-05
Q24	1.951852	1.951427	1.953964	2.5371E-03
Q510	-7.081546	-7.117440	-6.934647	-1.8279E-01

5.3.3 Example 3: Base case with inverted measurement error in line 6 (3-9)

This final example, for the AC model, is going to test how the MCC deals with a measurement value inversion. A different line measurement P_{39} (line 6) will be inverted. This was needed, because tests were done and if we used line 8 (34MW to -34MW) it greatly affected the RTSE results, thus making it converge to a non-feasible solution¹³. Since the EPSO's starting values (voltage module and angle) are given by the RTSE's results, it didn't allow the EPSO to converge in a timely and satisfactory manner. Therefore, line 6 was used as measurement instead of line 8 (used in the previous 2 examples).

Initially, we tried using the EPSO with a Parzen window size $\sigma=0.2$ just like in the previous examples. However, as can be checked in Annex B the results were not acceptable. Thus, after some tests a new Parzen window $\sigma=0.5$ was chosen, keeping the learning parameter the same value as before ($\tau=0.1$). By increasing the window size, one is able to make the objective (fitness) function smoother and also include all the "small" errors. This also allows for the EPSO to converge more easily to an optimal solution, faster and without any disadvantage on isolating the gross errors.

Once again, by looking at Figure 5.19 it is easy to see the evolution in the errors when comparing the MSE criterion and MCC's results. The MCC has similar behaviour as in previous examples, in that it showed a high degree of insensitiveness to the gross error, confirming its mentioned L0 metric properties [40]. Moreover, it also confirmed its L2 and L1 metric properties [40] by compressing the remaining errors close to zero, by effectively minimizing their variance, just like the MSE criterion does only the latter also tries to minimize the gross errors, thus contaminating the remaining errors as well.

Figure 5.20 represents the same graph as Figure 5.19 except for the gross measurement P_{39} . The MCC errors' evolution is remarkable, as one can see that the MAE value for the MSE criterion is 0.234307 while the MCC's value is 0.005085; a -0.229222 difference!

Another interesting result was the value for the gross error using the MCC. Even though the MSE criterion also tried correcting the gross error value to a somewhat close seed value, it also affected the remaining errors, for e.g. P_3 and P_9 (the extremities of the line 3-9) and the other injected active power nodes as well. On the other hand, the MCC corrected the gross error much closer to the seed value, with an error of only 0.131240. And by correcting that error value, it also minimized all the remaining errors with no exception. More information about the values can be consulted in Table 5.6.

¹³ It should be noted that the RTSE program was modified to make it not remove (what it thinks are) the gross error measurements, by the alternate probabilistic functions that deal with this issue. The reason for it was the need to have the gross measurements present, to be able to prove the MCC's ability to "ignore" those and minimize the remaining errors.

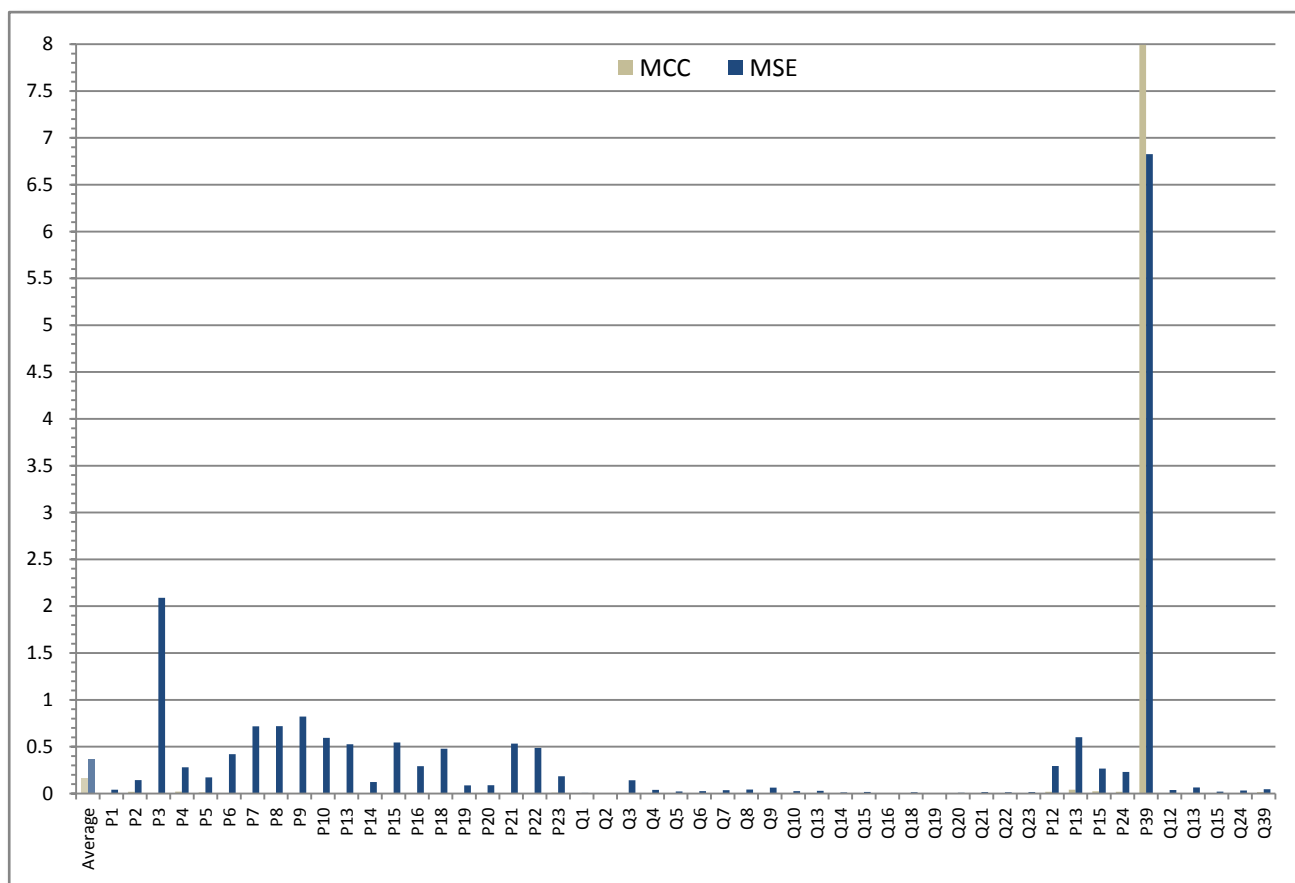


Figure 5.19 - Line 6 with error inverted: absolute errors and MAE. The y-axis represents the absolute errors (in MW/Mvar). The x-axis represents the 50 measurements considered and MAE ("Average").

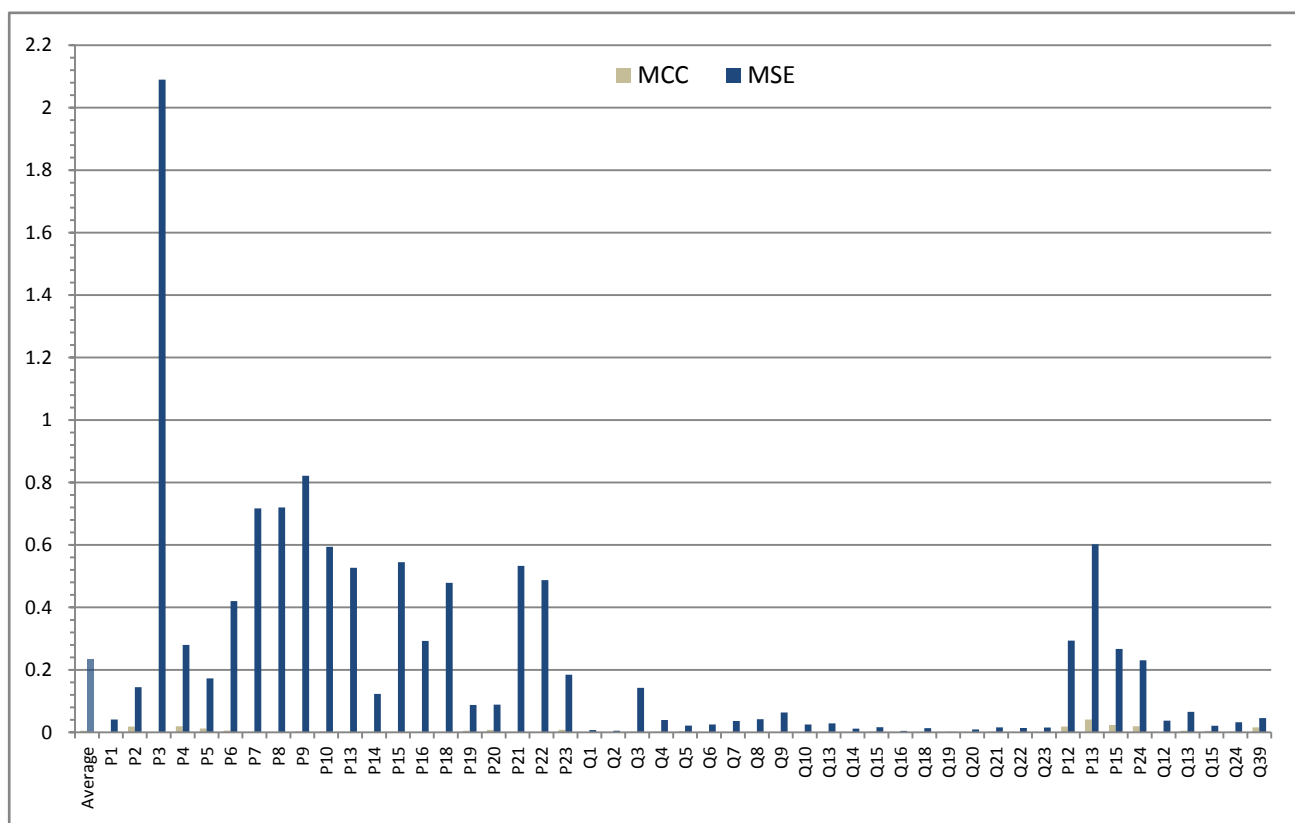


Figure 5.20 - Same graph as in Figure 5.19 excluding line 6 gross error. MAE recalculated.

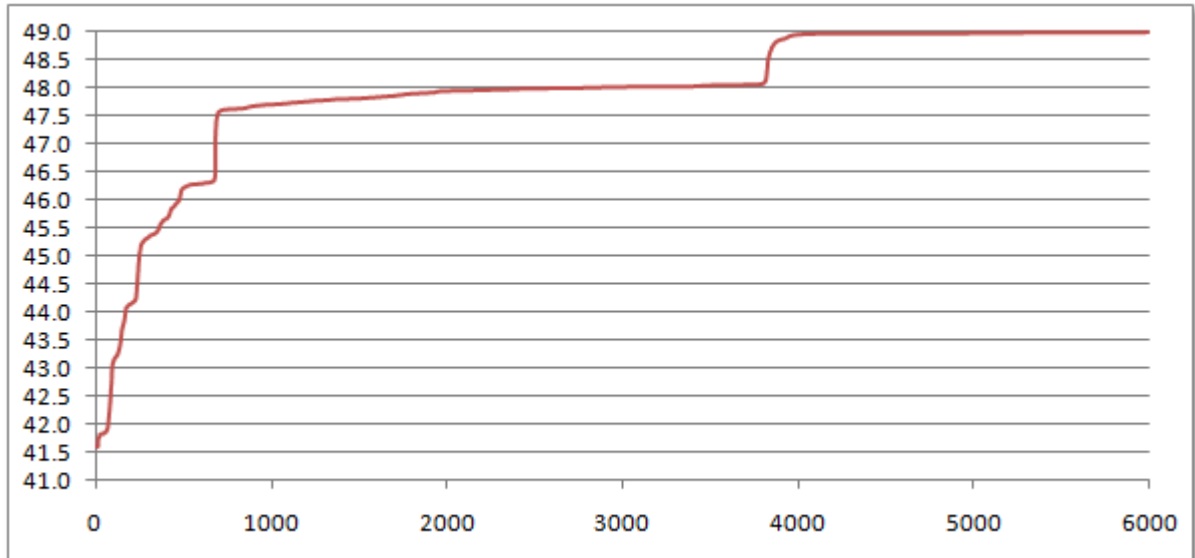


Figure 5.21 - EPSO fitness correntropy function evolution of example 3. The y-axis represents the correntropy function value. The x-axis represents the number of iterations.

In Figure 5.21, one can evaluate the evolution of the 10 run average of the EPSO through the 6000 iterations. This time, the initial correntropy value is around 41.5, which is a bit higher than in the previous examples. This is caused by the used Parzen window size value that is now 0.5 instead of 0.2. The wider the Parzen window size, the higher the starting correntropy value, up to a point where the MCC evolution would give results identical to the MSE criterion.

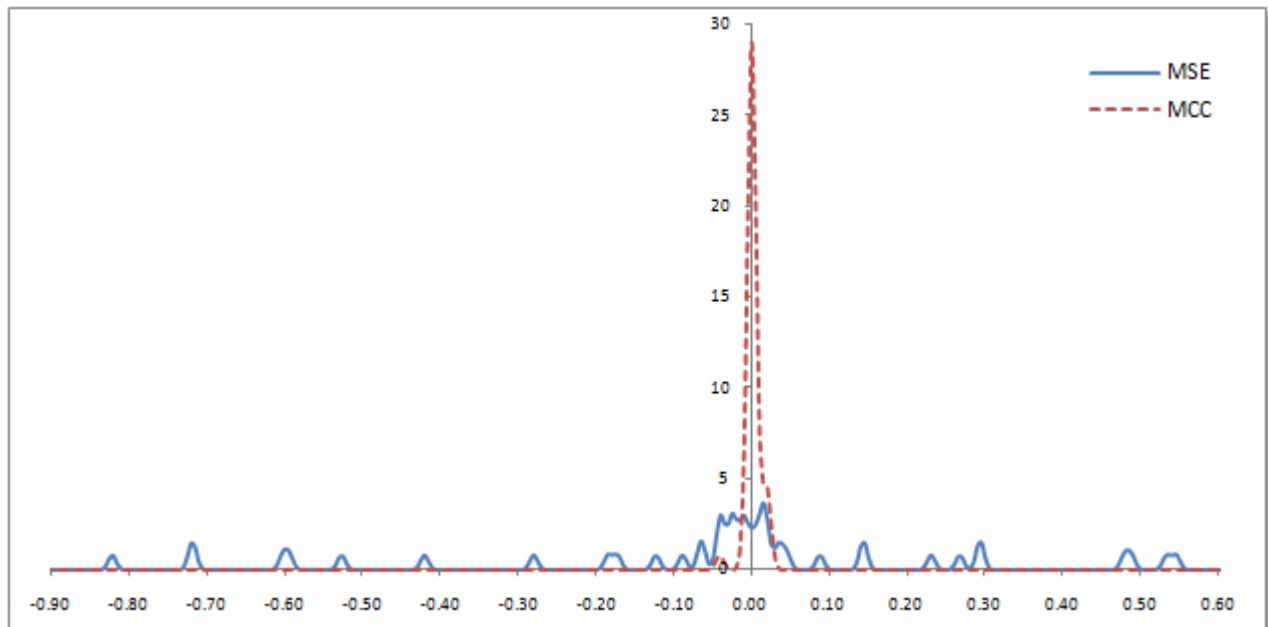


Figure 5.22 - Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.005$. The x-axis represents error values in MW/Mvar.

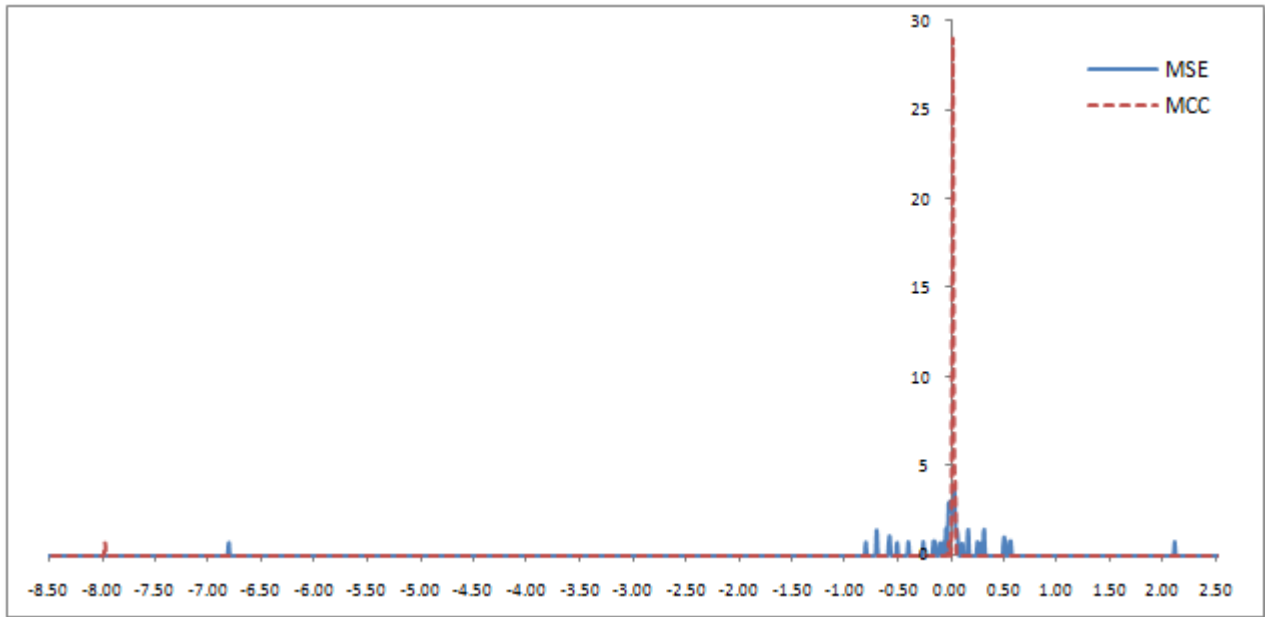


Figure 5.23 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.005$. The x-axis represents error values in MW/Mvar.

Like in previous examples, Figure 5.22 and Figure 5.23 represent the Gaussian distribution of the measurement errors, differing only in the inclusion of all the errors on the latter for a broader view, including the outlier isolated at around -8.0. A small Parzen window size is used $\sigma'=0.005$, in order to allow the observation of the location of the errors. It's apparent, when looking at Figure 5.23 that the MSE spread the errors quite a bit around the zero, unable to effectively minimize them (affected by the gross error). On the other hand, the MCC compressed all errors around zero and kicked the outlier further than the MSE criterion.

The next two figures represent the same tests, with different window scaling parameter sizes to illustrate how its changes affect the graphics' error distribution.

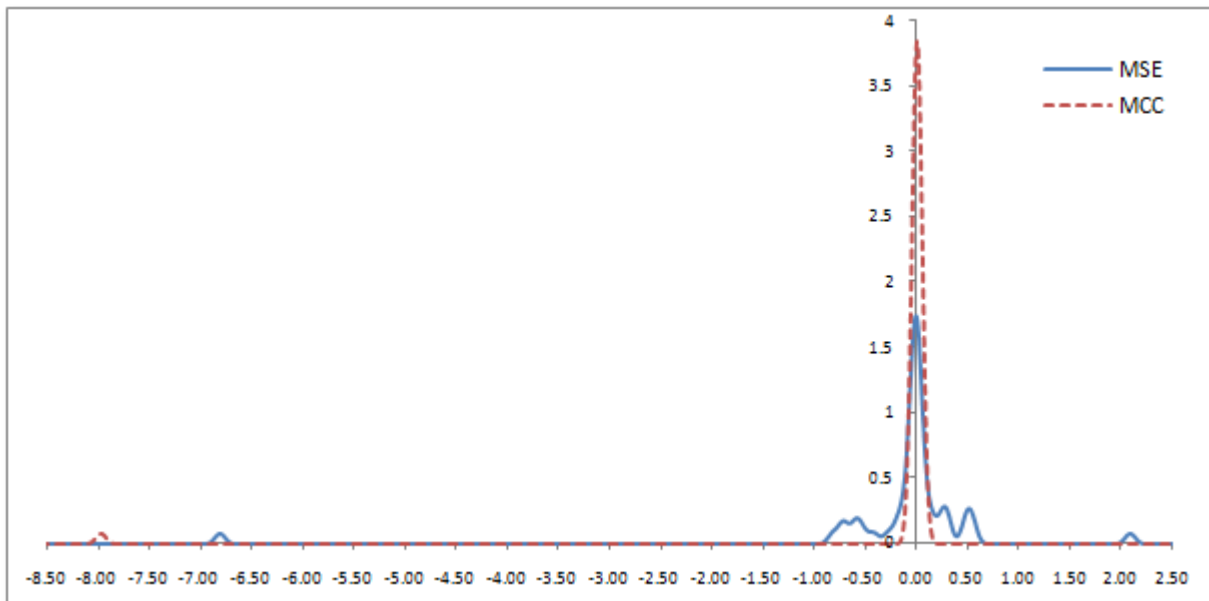


Figure 5.24 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.05$. The x-axis represents error values in MW/Mvar.

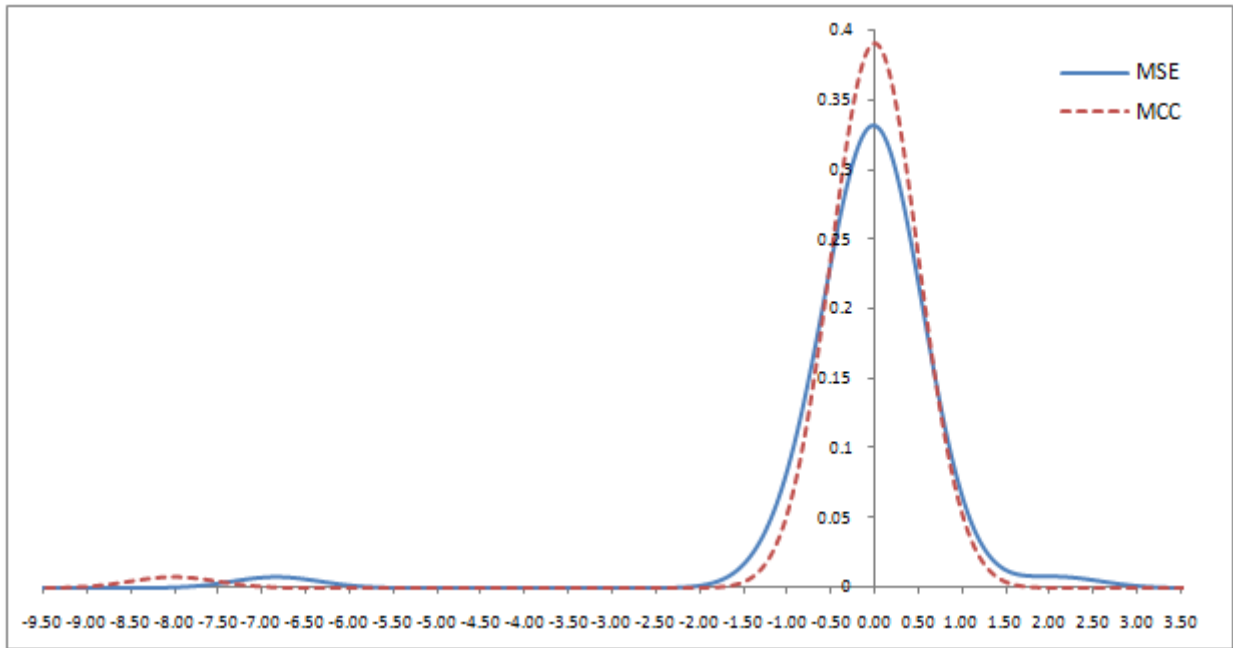


Figure 5.25 - Full Gaussian error distribution of example 3 using a Parzen window size $\sigma=0.5$ and $\sigma'=0.5$. The x-axis represents error values in MW/Mvar.

Figure 5.24 and Figure 5.25, use larger Parzen window scaling parameters to show the changes in the errors distributions. The representation of the error distribution allows us to see that, when using a narrow Parzen window scaling parameter, it doesn't "catch" all the small errors.

Therefore, it wouldn't be able to effectively optimize the fitness function. Furthermore, as one increases the window size scaling parameter, more errors are included. In Figure 5.23, it's easy to see that the MSE errors are concentrated mostly between -1 and 1, one error around 2.1 and the gross error at around -6.8. Parzen window size $\sigma'=0.05$ is still very narrow and does only include the closest errors to zero. By widening it more to $\sigma'=0.5$, all errors are now effectively included except for the outlier, as intended.

In Table 5.6 the values of the seed, initial measurements vector and respective errors used by the MCC the MSE criterion are shown. If we compare the MCC vs. MSE values, one sees that MCC was better in all of them except for the gross error measurement P_{39} . This can be explained by the fact that MCC's progression tries to "correct" the bad measurement to the most coherent values possible. This means that, if one compares the value of the estimated value of MCC against the seed, it's apparent that the difference between them is just 0.1312, while for the MSE criterion the value is 1.2974. It should be noted that the main objective of using the MCC is not to make the gross error the same value as the seed, but to minimize all the remaining errors variance without being affected by the gross error. The fact that the MCC also corrects the gross error to the most coherent value is a "bonus".

Finally, by looking at Table 5.7 one sees that, no matter the gross error value entered in the 3 examples, the final estimated MCC values were always very similar. This helps to prove the robustness and advantages of the MCC over the traditional methods currently used.

Table 5.6 - Seed, initial measurement errors and estimated values for example 3.

	Measurements			Errors		Estimated Values		Differences	
	ID	Seed	Error added	MSE	MCC	MSE	MCC	MCC vs MSE	MCC vs Seed
Injected Powers	P1	150.000000	150.000000	-4.12E-02	1.07E-03	150.041213	149.998934	-0.040147	-1.066E-03
	P2	10.000000	10.000000	1.44E-01	1.83E-02	9.855693	9.981723	-0.126029	-1.828E-02
	P3	-44.205356	-44.205356	2.09E+00	-1.04E-04	-46.295212	-44.205252	-2.089751	-1.044E-04
	P4	-18.255175	-18.255175	-2.80E-01	1.98E-02	-17.975203	-18.274932	-0.260215	1.976E-02
	P5	-17.436557	-17.436557	-1.73E-01	1.21E-02	-17.263986	-17.448702	-0.160426	1.214E-02
	P6	-33.399602	-33.399602	-4.20E-01	5.78E-03	-32.979266	-33.405384	-0.414554	5.782E-03
	P7	10.000000	10.000000	-7.17E-01	-1.44E-03	10.716815	10.001438	-0.715377	1.438E-03
	P8	-41.995088	-41.995088	-7.20E-01	1.28E-03	-41.275254	-41.996370	-0.718553	1.282E-03
	P9	-42.977430	-42.977430	-8.21E-01	3.21E-03	-42.156013	-42.980644	-0.818202	3.215E-03
	P10	-47.889136	-47.889136	-5.94E-01	-2.61E-03	-47.295118	-47.886528	-0.591410	-2.608E-03
	P13	10.000000	10.000000	-5.27E-01	-1.60E-03	10.526860	10.001596	-0.525264	1.596E-03
	P14	-47.643550	-47.643550	-1.23E-01	4.10E-04	-47.520705	-47.643961	-0.122436	4.102E-04
	P15	10.000000	10.000000	5.45E-01	1.68E-03	9.455104	9.998323	-0.543219	-1.677E-03
	P16	10.000000	10.000000	2.93E-01	-8.29E-04	9.707299	10.000829	-0.291872	8.294E-04
	P18	10.000000	10.000000	4.78E-01	9.23E-04	9.521511	9.999077	-0.477566	-9.233E-04
	P19	-44.450941	-44.450941	8.75E-02	-5.00E-03	-44.538480	-44.445942	-0.082540	-4.999E-03
	P20	-31.434920	-31.434920	-8.88E-02	-7.56E-03	-31.346161	-31.427358	-0.081196	-7.562E-03
	P21	10.000000	10.000000	5.33E-01	8.98E-04	9.467088	9.999102	-0.532014	-8.976E-04
	P22	10.000000	10.000000	4.87E-01	1.06E-03	9.512625	9.998940	-0.486316	-1.060E-03
	P23	143.534061	143.534061	-1.84E-01	-8.29E-03	143.718413	143.542351	-0.176062	8.290E-03
	Q1	-11.491917	-11.491917	-7.07E-03	-7.31E-04	-11.484845	-11.491186	-0.006342	-7.308E-04
	Q2	16.142114	16.142114	5.35E-03	-1.10E-03	16.136768	16.143216	-0.004244	1.102E-03
	Q3	-14.529598	-14.529598	1.42E-01	-2.10E-03	-14.671876	-14.527501	-0.140181	-2.097E-03
	Q4	-6.000186	-6.000186	-3.95E-02	1.87E-03	-5.960687	-6.002057	-0.037627	1.871E-03
	Q5	-5.731119	-5.731119	-2.15E-02	3.78E-03	-5.709634	-5.734895	-0.017709	3.776E-03
	Q6	-10.977918	-10.977918	-2.49E-02	-8.58E-04	-10.953046	-10.977060	-0.024014	-8.580E-04
	Q7	20.800598	20.800598	-3.64E-02	1.56E-04	20.836983	20.800442	-0.036229	-1.562E-04
	Q8	-13.803118	-13.803118	-4.22E-02	-4.50E-04	-13.760910	-13.802668	-0.041758	-4.498E-04
	Q9	-14.125998	-14.125998	-6.36E-02	1.14E-03	-14.062351	-14.127139	-0.062506	1.141E-03
	Q10	-15.740398	-15.740398	-2.50E-02	5.03E-05	-15.715373	-15.740448	-0.024974	5.030E-05
	Q13	46.060811	46.060811	-2.85E-02	-1.81E-03	46.089265	46.062619	-0.026646	1.808E-03
	Q14	-15.659678	-15.659678	-1.18E-02	4.69E-04	-15.647879	-15.660147	-0.011329	4.690E-04
	Q15	16.787847	16.787847	1.65E-02	2.82E-03	16.771306	16.785028	-0.013722	-2.819E-03
	Q16	38.978973	38.978973	4.37E-03	3.23E-04	38.974599	38.978650	-0.004051	-3.231E-04
	Q18	-1.262795	-1.262795	1.33E-02	-4.24E-04	-1.276102	-1.262371	-0.012882	-4.244E-04
	Q19	-14.610318	-14.610318	-1.68E-03	-4.22E-04	-14.608639	-14.609896	-0.001257	-4.222E-04
	Q20	-10.332159	-10.332159	-9.35E-03	-1.02E-03	-10.322804	-10.331136	-0.008332	-1.022E-03
	Q21	-1.492295	-1.492295	1.60E-02	2.35E-03	-1.508284	-1.494645	-0.013639	2.350E-03
	Q22	-1.279603	-1.279603	1.38E-02	1.15E-03	-1.293392	-1.280758	-0.012635	1.155E-03
	Q23	15.323757	15.323757	-1.53E-02	-6.22E-04	15.339021	15.324379	-0.014642	6.220E-04
Line Power flows	P12	62.912595	62.912595	2.94E-01	1.85E-02	62.618903	62.894082	-0.275179	-1.851E-02
	P13	34.079340	34.079340	-6.02E-01	-4.12E-02	34.681494	34.120578	-0.560916	4.124E-02
	P15	53.008065	53.008065	2.67E-01	2.38E-02	52.740816	52.984274	-0.243457	-2.379E-02
	P24	37.936167	37.936167	2.30E-01	1.95E-02	37.705720	37.916652	-0.210932	-1.952E-02
	P39	4.062118	-4.062118	-6.83E+00	-7.99E+00	2.764687	3.930878	1.166192	-1.312E-01
	Q12	-11.496704	-11.496704	3.75E-02	9.35E-05	-11.534176	-11.496798	-0.037378	9.352E-05
	Q13	-1.017736	-1.017736	-6.58E-02	-5.31E-03	-0.951986	-1.012424	-0.060439	-5.312E-03
	Q15	1.022523	1.022523	2.12E-02	4.49E-03	1.001317	1.018036	-0.016719	-4.487E-03
	Q24	1.951852	1.951852	3.22E-02	2.84E-03	1.919666	1.949017	-0.029351	-2.835E-03
	Q39	-2.043646	-2.043646	4.54E-02	1.58E-02	-2.089082	-2.059437	-0.029645	1.579E-02

Table 5.7 - Comparison between MCC results for example 1 and 3.

Measurements			MCC Results		Comparison
ID	Seed (Ex 1)	Seed (Ex 3)	Example 1	Example 3	Ex. 1 vs Ex. 3
P1	150.000000	150.000000	149.997716	149.998934	1.2183E-03
P2	10.000000	10.000000	9.995297	9.981723	-1.3574E-02
P3	-44.205356	-44.205356	-44.205525	-44.205252	-2.7341E-04
P4	-18.255175	-18.255175	-18.260519	-18.274932	1.4413E-02
P5	-17.436557	-17.436557	-17.440080	-17.448702	8.6217E-03
P6	-33.399602	-33.399602	-33.401556	-33.405384	3.8283E-03
P7	10.000000	10.000000	9.999484	10.001438	1.9539E-03
P8	-41.995088	-41.995088	-41.995472	-41.996370	8.9853E-04
P9	-42.977430	-42.977430	-42.977814	-42.980644	2.8305E-03
P10	-47.889136	-47.889136	-47.889290	-47.886528	-2.7619E-03
P13	10.000000	10.000000	9.999541	10.001596	2.0552E-03
P14	-47.643550	-47.643550	-47.644228	-47.643961	-2.6735E-04
P15	10.000000	10.000000	9.999953	9.998323	-1.6301E-03
P16	10.000000	10.000000	9.999797	10.000829	1.0323E-03
P18	10.000000	10.000000	9.999816	9.999077	-7.3892E-04
P19	-44.450941	-44.450941	-44.452225	-44.445942	-6.2828E-03
P20	-31.434920	-31.434920	-31.436843	-31.427358	-9.4859E-03
P21	10.000000	10.000000	9.999848	9.999102	-7.4523E-04
P22	10.000000	10.000000	9.999891	9.998940	-9.5025E-04
P23	143.534061	143.534061	143.532328	143.542351	1.0023E-02
Q1	-11.491917	-11.491917	-11.498939	-11.491186	-7.7532E-03
Q2	16.142114	16.142114	16.127609	16.143216	1.5607E-02
Q3	-14.529598	-14.529598	-14.529981	-14.527501	-2.4802E-03
Q4	-6.000186	-6.000186	-6.007105	-6.002057	-5.0480E-03
Q5	-5.731119	-5.731119	-5.752319	-5.734895	-1.7423E-02
Q6	-10.977918	-10.977918	-10.981316	-10.977060	-4.2560E-03
Q7	20.800598	20.800598	20.800281	20.800442	1.6120E-04
Q8	-13.803118	-13.803118	-13.803156	-13.802668	-4.8791E-04
Q9	-14.125998	-14.125998	-14.126725	-14.127139	4.1395E-04
Q10	-15.740398	-15.740398	-15.740773	-15.740448	-3.2483E-04
Q13	46.060811	46.060811	46.060660	46.062619	1.9590E-03
Q14	-15.659678	-15.659678	-15.659828	-15.660147	3.1845E-04
Q15	16.787847	16.787847	16.787829	16.785028	-2.8010E-03
Q16	38.978973	38.978973	38.978788	38.978650	-1.3831E-04
Q18	-1.262795	-1.262795	-1.262927	-1.262371	-5.5596E-04
Q19	-14.610318	-14.610318	-14.610831	-14.609896	-9.3566E-04
Q20	-10.332159	-10.332159	-10.332650	-10.331136	-1.5141E-03
Q21	-1.492295	-1.492295	-1.492574	-1.494645	2.0708E-03
Q22	-1.279603	-1.279603	-1.279943	-1.280758	8.1491E-04
Q23	15.323757	15.323757	15.323341	15.324379	1.0385E-03
P12	62.912595	62.912595	62.909941	62.894082	-1.5859E-02
P13	34.079340	34.079340	34.073197	34.120578	4.7381E-02
P15	53.008065	53.008065	53.002292	52.984274	-1.8019E-02
P24	37.936167	37.936167	37.931814	37.916652	-1.5162E-02
P510/P39	34.958731	4.062118	34.982992	3.930878	-
Q12	-11.496704	-11.496704	-11.505835	-11.496798	-9.0373E-03
Q13	-1.017736	-1.017736	-1.020813	-1.012424	-8.3889E-03
Q15	1.022523	1.022523	1.021554	1.018036	-3.5182E-03
Q24	1.951852	1.951852	1.951427	1.949017	-2.4105E-03
Q510/Q39	-7.081546	-2.043646	-7.117440	-2.059437	-

5.4 Chapter conclusions

At the end of this chapter, one was able to confirm the results obtained for the DC model (Chapter 4). Therefore, we were able to test both DC and AC models successfully and effectively prove the concept that using the maximum correntropy criterion (MCC) it's possible to obtain a better solution in the presence of gross measurements without the need to remove those. By being able to successfully improve the MSE criterion's solutions in a much more complex AC 24-bus network than the 4-bus DC model of Chapter 4, the MCC proved that it can be an alternative to the currently used regression estimator algorithms like the MSE criterion. It should also be mentioned that an evolutionary algorithm like the EPSO is not, by far, an ideal method for a real-time state estimator as it has slow convergence. However, it was sufficient to perform the AC tests successfully. To be able to use the EPSO, there was a need to do several tests to choose the best externally defined parameters such as the cooperation probability " p " and the learning parameter " τ ". After all the testing, the optimal values found were $\sigma=0.2$ and $\tau=0.1$, however they had to be changed in a few occasions to be able to converge successfully when in the presence of high magnitude gross errors, where the MSE's results distortion was bigger.

Furthermore, one shouldn't forget about the importance of the Parzen window size " σ " in the modulating effect of the MCC over the MSE, or in other words, the boundaries where the MCC behaves as a L2 metric or as a L0 metric. In the various presented examples, either in AC or DC, it was apparent the importance of choosing a good Parzen window size for the best optimization possible.

In **example 1**, a +5% measurement error (+1.75MW) was tested to see how the maximum correntropy criterion behaved on the presence of a low magnitude gross error. The test was successful and, like in what happened on the DC model tests, the MCC was able to isolate the gross error P_{510} and compressed the remaining measurement errors close to zero.

For **example 2**, the idea was to confirm the L0 metric properties of the MCC in the AC model by increasing the same measurement error value (P_{510}), but this time by +20% (6.992MW). Once again, the property was verified, since the gross measurement error was again isolated and the MCC compressed the remaining errors. It didn't affect the outcome, the fact of increasing the gross error's magnitude. However, there was a need to increase the Parzen window size to **0.4** to be able to include all the remaining (non-gross) errors from the MSE criterion results.

Finally in **example 3**, a measurement error was inverted, but in this case there was a need to use a different line measurement than line 8 (5-10). This was due to the bad behaviour of the RTSE under the test of that line. Since the EPSO receives the solution from the RTSE and that solution was very poor, there was not a feasible solution to optimize. Thus, it was decided to use line 6 (3-9) instead, as this is a line with lower active power value thus less likely to cause a large disturbance in the RTSE's solution. In this test, there was also a need to change the Parzen window size to **0.5** for the same reasons as reported before. The final MCC estimated values were very good, as shown in Table 5.6.

A comparison of the 3 example's (Table 5.5 and Table 5.7) behaviour against the seed shows how approximate the MCC's results were to the seed's. This is the intended result, thus proving the concept, as described and shown throughout this Thesis.

Chapter 6

Conclusions

6.1 General Conclusions

This Thesis provided and tested a new concept on state estimation, using a correntropy optimization function. Tests were made using a 4-bus DC network and a more complex 24-bus network AC models approach, in order to test the known properties of the correntropy functions, namely the L1/L2 metric (variance) for small errors and a L0 metric on gross errors.

In relation to the DC tests, although a very simple network was used, they gave promising results and imposed the gross measurement error (outlier) isolation, keeping the remaining errors almost unaffected.

With the AC model experiments, the objective was to confirm the obtained results for the DC model, thus validating its use on this more complex formulation.

As explained in more detail in Chapter 4 and 5's conclusions, the tests were successful in proving that better solutions to the traditional MSE (WLS) method do exist. The author was able to verify that when gross errors are not present, the correntropy state estimation results are identical to the ones given using the MSE (WLS) method, thus effectively confirming the L2 metric, i.e. minimizing the error's variance the same way as the MSE. On the other hand, when gross errors were present the MSE state estimator contaminated the remaining errors, therefore giving poorer results, as expected; while the MCC successfully isolates the gross measurement error and compresses the remaining (smaller) errors around the zero mean of the Gaussian distribution.

One interesting result was when using an inverted line measurement error, which is an error that happens occasionally in the power systems. One case where the error was inverted passing from 4 MW to -4 MW the maximum correntropy criterion (MCC) was able to isolate that measurement as outlier and compressed the remaining errors very close to zero (unlike the MSE) and furthermore corrected, almost completely, the gross measurement to its seed value. This is a very interesting result, which demonstrates how the MCC tries to "extract" the most coherent solution possible from the whole errors' information samples.

6.2 Future Work

Since this Thesis' point was to prove the concept of correntropy properties and confirm the existence of alternative (better) solutions to the state estimation algorithms, there are now a wide number of future ideas that need researching and to be worked on. The following future developments are suggested by the author:

- The nature of this Thesis implied the use of a meta-heuristic algorithm, like the EPSO and that's because we were interested in obtaining several optimums on the search of the best value possible, therefore searching the domain space and look for alternative feasible solutions. If, for example a method like the gradient was used, one would obtain the same solution every time (if the same conditions were used) and this would greatly limit the experience results. But, now that one knows that such (better) solutions exist, the next step will be to take advantage of the correntropy functions being differentiable and, thus, can be used, after the mathematical deduction and formulation, by a much faster algorithm like a Newton-Raphson. This would allow the correntropy concept to be able to compete with commercial applications currently used;
- Because only nodal injected power and line power flow measurements were used, it would be interesting to test how the including additional measurements such as voltage, would affect the outlier identification and optimization of the remaining errors;
- Another area that should be analysed is the one related to topology processing, because if it isn't good enough, and doesn't provide an accurate snapshot of the network topology the state estimation results can be way off, from the expected values. Furthermore, it's known that the error distribution of the topology measurements is not (by far, in most cases) Gaussian. That means that the WLS approach won't be the best estimator for them. Therefore, using the maximum correntropy criterion (MCC) it should be possible to get better results;
- More thorough testing needs to be done related to the Parzen window sizes. It is important to check how it affects the MCC convergence and results, outlier isolation and remaining errors compression;
- Somewhat related to the previous one, a concept that deserves more attention is the Parzen window size annealing. Because of time constraints, it wasn't possible to effectively test this concept, therefore is not presented. However, it should prove a good research topic since, as it was verified during this work, the variation of the window size can have dramatic changes in the graph's shape, allowing more or less local optimums. A possible methodology, that's been successfully tested in other cases, is to use a wide (large) window in the beginning of the optimization process, so that the curves are smooth and the algorithm is able to quickly evolve to the maximum objective function value (which is in the zero mean of the Gaussian distribution) without getting stuck. As the process evolves, the window starts getting

smaller and smaller in order to start isolating the outliers and compress the remaining errors. Also, a more intelligent approach can be used, for example, using a self-adaptive Parzen window size that increases and decreases as needed, for getting best results;

- There needs to be a more complex analysis of the behaviour of the correntropy estimator, for example, by adding more random errors to the measurements to see how they affect the outcome and if they are correctly optimized, like they would be on a WLS algorithm after the bad measurements removal;
- One limitation of the tests made in this Thesis was the starting values (voltage module and angle) used by the EPSO. Those were extracted from the traditional real-time state estimator (RTSE) solution. The problem is that when gross measurement errors were added and the estimator didn't remove the bad measurements (on purpose), the voltage values solution was much distorted and probably didn't allow an optimal solution, as was desired. Again, some tests were done, including a flat start and a randomization of the starting values, but there was no time to do enough experiments in order to conclude anything on this issue. More tests need to be done on the starting values because the author believes that, those will be important when using a classic numerical optimization algorithm such as the Newton-Raphson.

References

- [1] "IEEE Standard Letter Symbols for Units of Measurement (SI Units, Customary Inch-Pound Units, and Certain Other Units)", IEEE Std 260.1-2004 (Revision of IEEE Std 260.1-1993), (2004), pp. 23.
- [2] Sasson, AM, Ehrmann, ST, Lynch, P, and Van Slyck, LS, "Automatic Power System Network Topology Determination". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-92, Issue: 2 (1973), pp. 610-618.
- [3] Krumpholz, G.R., "Electric power network measurement system design". Thesis (Ph. D.), Worcester Polytechnic Institute, Worcester, MA, United States, 1981, pp. 165.
- [4] Monticelli, A., and Wu, F.F., "Network observability: theory". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-104, Issue: 5 (1985), pp. 1042-1048.
- [5] Schweppe, FC, and Wildes, J, "Power System Static-State Estimation, Part I: Exact Model". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-89, Issue: 1 (1970), pp. 120-125.
- [6] Schweppe, FC, and Rom, DB, "Power System Static-State Estimation, Part II: Approximate Model". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-89, Issue: 1 (1970), pp. 125-130.
- [7] Schweppe, FC, "Power System Static-State Estimation, Part III: Implementation". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-89, Issue: 1 (1970), pp. 130-135.
- [8] Dopazo, JF, Klitin, OA, Stagg, GW, and Van Slyck, LS, "State Calculation of Power Systems From Line Flow Measurements". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-89, Issue: 7 (1970), pp. 1698-1708.
- [9] Garcia, A, Monticelli, A, and Abreu, P, "Fast Decoupled State Estimation and Bad Data Processing". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-98, Issue: 5 (1979), pp. 1645-1652.
- [10] Handschin, E, Schweppe, FC, Kohlas, J, and Fiechter, A, "Bad data analysis for power system state estimation". IEEE Transactions on Power Apparatus and Systems, Vol: 94, Issue: 2, Part 1 (1975), pp. 329-337.
- [11] Merrill, HM, and Schweppe, FC, "Bad Data Suppression in Power System Static State Estimation". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-90, Issue: 6 (1971), pp. 2718-2725.
- [12] Monticelli, A, and Garcia, A, "Reliable Bad Data Processing for Real-Time State Estimation". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-102, Issue: 5 (1983), pp. 1126-1139.
- [13] Van Cutsem, T, Ribbens-Pavella, M, and Mili, L, "Bad Data Identification Methods In Power System State Estimation-A Comparative Study". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-104, Issue 11 (1985), pp. 3037-3049.
- [14] Abur, Ali, and Expósito, António Gómez, "Power System State Estimation: Theory and Implementation", CRC Press, (2004), pp. 327.
- [15] REN - Redes Energéticas Nacionais, <http://www.ren.pt/> (2008)

- [16] Pereira, JMC, "A State Estimation Approach for Distribution Networks Considering Uncertainties and Switching". Thesis (Ph. D.), Faculty of Engineering of the University of Porto, Portugal, 2001, pp. 233.
- [17] Monticelli, A, "State Estimation in Electric Power Systems: A Generalized Approach", Kluwer Academic Publishers, (1999), pp. 394.
- [18] Xu, B, and Abur, A, "Optimal Placement of Phasor Measurement Units for State Estimation". Final Project Report, Texas A&M University, October 2005.
- [19] Phadke, A. G., "Synchronized phasor measurements-a historical overview". Transmission and Distribution Conference and Exhibition 2002: Asia Pacific. IEEE/PES, pp. 476-479 (vol.1)
- [20] Yu-Qing, Jin, Chuan, Qin, Feng, Wu, Jing-Dong, Han, Qun, Xu, Deng-Jun, Yan, and Ping, Ju, "The power Angle and Phase Measurement Units based Wide Area Measurement System and its application". Bulk Power System Dynamics and Control - VII. Revitalizing Operational Reliability, 2007 iREP Symposium, pp. 1-7
- [21] Bose, A, "The effects of external modeling errors on on-line security analysis". IEEE Transactions on Power Systems, Vol: 1, Issue: 1 (1986), pp. 227-232.
- [22] Deckmann, S., Pizzolante, A., Monticelli, A., Stott, B., and Alsac, O., "Numerical Testing of Power System Load Flow Equivalents". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-99 (1980), pp. 2292-2300.
- [23] Deckmann, S., Pizzolante, A., Monticelli, A., Stott, B., and Alsac, O., "Studies on Power System Load Flow Equivalencing". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-99 (1980), pp. 2301-2310.
- [24] Bose, A., "Modeling of External Networks for On-Line Security Analysis". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-103, Issue: 8 (1984), pp. 2117-2125.
- [25] Geisler, K. I., and Bose, A., "State Estimation Based External Network Solution For On-Line Security Analysis". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-102, Issue: 8 (1983), pp. 2447-2454.
- [26] Geisler, K. I., and Tripathi, N. K., "State Estimation Based External Network Solutions: Field Testing". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-104, Issue: 8 (1985), pp. 2126-2132.
- [27] Monticelli, A., and Wu, F. F., "A Method That Combines Internal State Estimation and External Network Modeling". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-104 (1985), pp. 91-103.
- [28] Lu, C. N., Liu, K. C., and Vemuri, S., "An external network modeling approach for online security analysis". IEEE Transactions on Power Apparatus and Systems, Vol: 5 (1990), pp. 565-573.
- [29] Sanderson, P., Curtis, R., Athow, D., Lu, C. N., Liu, K. C., and Letter, C., "Real time complete model estimation for contingency study: field experience". IEEE Transactions on Power Systems, Vol: 6 (1991), pp. 1480-1484.
- [30] Korres, G. N., "A partitioned state estimator for external network modeling". IEEE Transactions on Power Apparatus and Systems, Vol: 17 (2002), pp. 834-842.
- [31] Van Cutsem, T, Ribbens-Pavella, M, and Mili, L, "Hypothesis Testing Identification: A New Method For Bad Data Analysis In Power System State Estimation". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-103 (1984), pp. 3239-3252.
- [32] Miranda, Vladimiro, "Entropia e Treino de Sistemas - Versão 1.0", (2005).
- [33] Shannon, CE, and Weaver, W, "A mathematical theory of communications". Bell System Technical Journal, Vol: 27 (1948), pp. 632-656.
- [34] Shannon, CE, and Weaver, W, "The Mathematical Theory of Communication", University of Illinois Press (1949).
- [35] Príncipe, JC, and Xu, D, "An introduction to information theoretic learning". International Joint Conference on Neural Networks [IJCNN '99], (1999), pp. 1783-1787 (vol.3)
- [36] Príncipe, JC, and Xu, D, "Information-theoretic learning using Renyi's quadratic entropy". Proceedings of the First International Workshop on Independent Component Analysis and Signal Separation, Aussois, (1999), pp. 407-412
- [37] Parzen, E, "On the estimation of a probability density function and mode". Annals of Mathematical Statistics, Vol: 33 (1962), pp. 1065-1076.
- [38] Renyi, A, "Some fundamental questions of information theory". Selected Papers of Alfred Renyi, Vol: 2 (1976), pp. 526-552.

- [39] Liu, W, Pokharel, PP, and Príncipe, JC, "Error Entropy, Correntropy and M-Estimation", Proceedings of the 2006 16th IEEE Signal Processing Society Workshop on Machine Learning for Signal Processing, (2006), pp. 179-184.
- [40] Liu, W, Pokharel, PP, and Príncipe, JC, "Correntropy: Properties and Applications in Non-Gaussian Signal Processing". IEEE Transactions on Signal Processing [see also IEEE Transactions on Acoustics, Speech, and Signal Processing], Vol: 55 (2007), pp. 5286-5298.
- [41] Santamaria, I, Pokharel, PP, and Principe, JC, "Generalized correlation function: definition, properties, and application to blind equalization". IEEE Transactions on Signal Processing [see also IEEE Transactions on Acoustics, Speech, and Signal Processing], Vol: 54 (2006), pp. 2187-2197.
- [42] Subcommittee, IEEE RTS Task Force of APM, "IEEE Reliability Test System". IEEE Transactions on Power Apparatus and Systems, Vol: PAS-98, No. 6 Nov./Dec. (1979), pp. 2047-2054.
- [43] Pereira, JC, Saraiva, JT, Miranda, V, Costa, AS, Lourenço, EM, and Clements, KA, "Comparison of approaches to identify topology errors in the scope of state estimation studies". Power Tech Proceedings, 2001 IEEE Porto, (2001), pp. 6
- [44] Pereira, J, Saraiva, JT, and Miranda, V, "An integrated load allocation/state estimation approach for distribution networks". International Conference on Probabilistic Methods Applied to Power Systems, (2004), pp. 180-185
- [45] Keko, H, Duque, AJ, and Miranda, V, "A Multiple Scenario Security Constrained Reactive Power Planning Tool Using EPSO". International Conference on Intelligent Systems Applications to Power Systems, (2007), pp. 1-6
- [46] Miranda, V., Keko, H., and Duque, A., "Stochastic Star Communication Topology in Evolutionary Particle Swarms (EPSO)". International Journal of Computational Intelligence Research, Vol: 4, Issue : 2 (2008), pp. 105-116.
- [47] Miranda, V, Keko, H, and Jaramillo, A, "EPSO: Evolutionary Particle Swarms". Advances in Evolutionary Computing for System Design, Vol: 66 (2007), pp. 139-167.
- [48] Goldberg, DE, "Genetic Algorithms in Search, Optimization and Machine Learning", Addison-Wesley Longman Publishing Co., Inc. Boston, MA, USA, (1989), pp. 432.
- [49] Gen, M, and Cheng, R, "Genetic Algorithms and Engineering Design", Wiley-Interscience, (1997), pp. 411.
- [50] Kennedy, J., and Eberhart, R., "Particle swarm optimization". IEEE International Conference on Neural Networks, 1995. Proceedings., pp. 1942-1948 (vol.4)
- [51] Stott, B, Alsac, O, and Monticelli, AJ, "Security analysis and optimization". Proceedings of the IEEE, Vol: 75 (1987), pp. 1623-1644.
- [52] UCTE - Union for the Co-ordination of Transmission of Electricity, <http://www.ucte.org/> (2008)
- [53] 5/11/2006, UCTE - News, <http://www.ucte.org/news/20061105/> (2006)
- [54] 6/11/2006, UCTE - News, <http://www.ucte.org/news/20061106/> (2006)
- [55] 16/11/2006, UCTE - News, <http://www.ucte.org/news/20061116/> (2006)
- [56] 30/11/2006, UCTE - News, <http://www.ucte.org/news/20061130/> (2006)
- [57] UCTE - Interim Report on the disturbances of 4 November, <http://www.ucte.org/library/news/IC-Interim-Report-20061130.pdf> (2006)
- [58] Dy Liacco, Thomas E., "Real-Time Computer Control of Power Systems". Proceedings of the IEEE, Vol: 62, Issue: 7 (1974), pp. 884-891.
- [59] Sollberger, MH, "Security of an Industrial System". Case Western Reserve University, 1971.

Annexes

Annex A

IEEE Reliability Test System (AC model)

A.1 Network topology

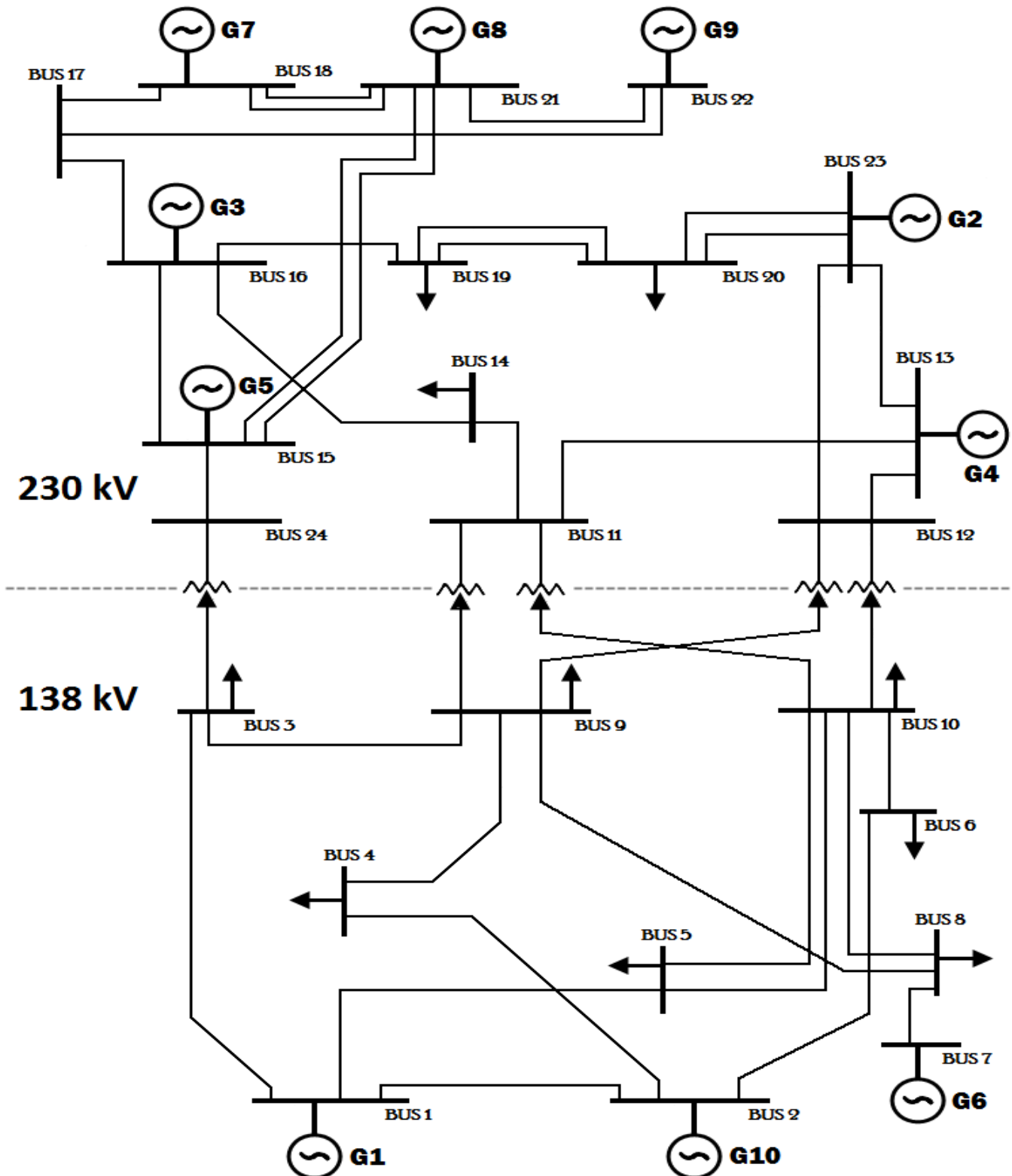


Figure A.1 - Network topology of the IEEE Reliability network with minor modifications.

A.2 Network parameters

Table A.1 - IEEE 24bus network line and transformer parameters ($S_b = 100\text{MVA}$).

From Bus	To Bus	Impedance (P.U./100 MVA Base)		Rating (MVA)	Equipment
		R	X		
1	2	0.00260	0.01390	175	138kV cable
1	3	0.05460	0.21120	175	138kV line
1	5	0.02180	0.08450	175	138kV line
2	4	0.03280	0.12670	175	138kV line
2	6	0.04970	0.19200	175	138kV line
3	9	0.03080	0.11900	175	138kV line
3	24	0.00000	0.08390	400	Transformer
4	9	0.02680	0.10370	175	138kV line
5	10	0.02280	0.08830	175	138kV line
6	10	0.01390	0.06050	175	138kV cable
7	8	0.01590	0.06140	175	138kV line
8	9	0.04270	0.16510	175	138kV line
8	10	0.04270	0.16510	175	138kV line
9	11	0.00000	0.08390	400	Transformer
9	12	0.00000	0.08390	400	Transformer
10	11	0.00000	0.08390	400	Transformer
10	12	0.00000	0.08390	400	Transformer
11	13	0.00610	0.04760	500	230kV line
11	14	0.00540	0.04180	500	230kV line
12	13	0.00610	0.04760	500	230kV line
12	23	0.01240	0.09660	500	230kV line
13	23	0.01110	0.08650	500	230kV line
14	16	0.00500	0.03890	500	230kV line
15	16	0.00220	0.01730	500	230kV line
15	21	0.00630	0.04900	500	230kV line
15	21	0.00630	0.04900	500	230kV line
15	24	0.00670	0.05190	500	230kV line
16	17	0.00330	0.02590	500	230kV line
16	19	0.00300	0.02310	500	230kV line
17	18	0.00180	0.01440	500	230kV line
17	22	0.01350	0.10530	500	230kV line
18	21	0.00330	0.02590	500	230kV line
18	21	0.00330	0.02590	500	230kV line
19	20	0.00510	0.03960	500	230kV line
19	20	0.00510	0.03960	500	230kV line
20	23	0.00280	0.02160	500	230kV line
20	23	0.00280	0.02160	500	230kV line
21	22	0.00870	0.06780	500	230kV line

Table A.2 - IEEE 24bus network load values.

LoadId	NodeId	Real Power (MW)	Reactive Power (Mvar)
load0003	node0003	44.20535610	14.52959791
load0004	node0004	18.25517483	6.000185804
load0005	node0005	17.43655713	5.731119178
load0006	node0006	33.39960239	10.97791842
load0008	node0008	41.99508829	13.80311802
load0009	node0009	42.97742954	14.12599797
load0010	node0010	47.88913577	15.74039774
load0012	node0014	47.64355046	15.65967775
load0016	node0019	44.45094141	14.61031790
load0017	node0020	31.43491989	10.33215852

Table A.3 - IEEE 24bus network synchronous generator parameters.

Sync. Gen. Id	Node Id	Max Active Power (MW)	Min Active Power (MW)	Nom. Voltage (kV)	Nom. App. Power (MVA)	Nom. Power Factor	Pos. Seq. Synchron. Impedance (Ω)	Neg. Seq. Impedance (Ω)	Neutral Earthing Impedance (Ω)	Active Power Dispatch (MW)
1	1	450	150	138	384	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	150
2	23	455	150	230	600	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	155*
3	16	130	20	230	1182	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
4	13	130	20	230	430	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
5	15	162	25	230	310	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
6	7	80	20	138	800	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
7	18	85	25	230	800	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
8	21	100	10	230	600	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
9	22	100	10	230	1320	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10
10	2	100	10	138	1320	0.95	$0.01+j12.2$	$0.01+j12$	$1000+j1000$	10

*slack bus (different value after powerflow)

Table A.4 - Measurements used for the seed values, 5% and 20% error on line 5-10.

	Measurement			
	Location	Seed Values (in MW, Mvar)	+5% active power error to line 5-10	+20% active power error to line 5-10
<i>Injected Power</i>	P1	150.000000	150.000000	150.000000
	P2	10.000000	10.000000	10.000000
	P3	-44.205356	-44.205356	-44.205356
	P4	-18.255175	-18.255175	-18.255175
	P5	-17.436557	-17.436557	-17.436557
	P6	-33.399602	-33.399602	-33.399602
	P7	10.000000	10.000000	10.000000
	P8	-41.995088	-41.995088	-41.995088
	P9	-42.977430	-42.977430	-42.977430
	P10	-47.889136	-47.889136	-47.889136
	P13	10.000000	10.000000	10.000000
	P14	-47.643550	-47.643550	-47.643550
	P15	10.000000	10.000000	10.000000
	P16	10.000000	10.000000	10.000000
	P18	10.000000	10.000000	10.000000
	P19	-44.450941	-44.450941	-44.450941
	P20	-31.434920	-31.434920	-31.434920
	P21	10.000000	10.000000	10.000000
	P22	10.000000	10.000000	10.000000
	P23	143.534061	143.534061	143.534061
	Q1	-11.491917	-11.491917	-11.491917
	Q2	16.142114	16.142114	16.142114
	Q3	-14.529598	-14.529598	-14.529598
	Q4	-6.000186	-6.000186	-6.000186
	Q5	-5.731119	-5.731119	-5.731119
	Q6	-10.977918	-10.977918	-10.977918
	Q7	20.800598	20.800598	20.800598
	Q8	-13.803118	-13.803118	-13.803118
	Q9	-14.125998	-14.125998	-14.125998
	Q10	-15.740398	-15.740398	-15.740398
	Q13	46.060811	46.060811	46.060811
	Q14	-15.659678	-15.659678	-15.659678
	Q15	16.787847	16.787847	16.787847
	Q16	38.978973	38.978973	38.978973
	Q18	-1.262795	-1.262795	-1.262795
	Q19	-14.610318	-14.610318	-14.610318
	Q20	-10.332159	-10.332159	-10.332159
	Q21	-1.492295	-1.492295	-1.492295
	Q22	-1.279603	-1.279603	-1.279603
	Q23	15.323757	15.323757	15.323757
<i>Power line flow</i>	P12	62.912595	62.912595	62.912595
	P13	34.079340	34.079340	34.079340
	P15	53.008065	53.008065	53.008065
	P24	37.936167	37.936167	37.936167
	P510	34.958731	36.706668	41.950477
	Q12	-11.496704	-11.496704	-11.496704
	Q13	-1.017736	-1.017736	-1.017736
	Q15	1.022523	1.022523	1.022523
	Q24	1.951852	1.951852	1.951852
	Q510	-7.081546	-7.081546	-7.081546

Table A.5 - Measurements used for the seed values and inverted error on line 3-9.

	Measurement		
	Location	Seed Values (in MW, Mvar)	Inverted active power of line 3-9
Injected Power	P1	150.000000	150.000000
	P2	10.000000	10.000000
	P3	-44.205356	-44.205356
	P4	-18.255175	-18.255175
	P5	-17.436557	-17.436557
	P6	-33.399602	-33.399602
	P7	10.000000	10.000000
	P8	-41.995088	-41.995088
	P9	-42.977430	-42.977430
	P10	-47.889136	-47.889136
	P13	10.000000	10.000000
	P14	-47.643550	-47.643550
	P15	10.000000	10.000000
	P16	10.000000	10.000000
	P18	10.000000	10.000000
	P19	-44.450941	-44.450941
	P20	-31.434920	-31.434920
	P21	10.000000	10.000000
	P22	10.000000	10.000000
	P23	143.534061	143.534061
	Q1	-11.491917	-11.491917
	Q2	16.142114	16.142114
	Q3	-14.529598	-14.529598
	Q4	-6.000186	-6.000186
	Q5	-5.731119	-5.731119
	Q6	-10.977918	-10.977918
	Q7	20.800598	20.800598
	Q8	-13.803118	-13.803118
	Q9	-14.125998	-14.125998
	Q10	-15.740398	-15.740398
	Q13	46.060811	46.060811
	Q14	-15.659678	-15.659678
	Q15	16.787847	16.787847
	Q16	38.978973	38.978973
	Q18	-1.262795	-1.262795
	Q19	-14.610318	-14.610318
	Q20	-10.332159	-10.332159
	Q21	-1.492295	-1.492295
	Q22	-1.279603	-1.279603
	Q23	15.323757	15.323757
Power line flow	P12	62.912595	62.912595
	P13	34.079340	34.079340
	P15	53.008065	53.008065
	P24	37.936167	37.936167
	P39	4.062118	-4.062118
	Q12	-11.496704	-11.496704
	Q13	-1.017736	-1.017736
	Q15	1.022523	1.022523
	Q24	1.951852	1.951852
	Q39	-2.043646	-7.081546

Annex B

Problem data and some results

B.1 AC - Example 3: Line 6 gross error non-feasible solution with $\sigma=0.2$

In this annex, the following graphs illustrate a case where the EPSO didn't converge to a feasible solution because of an inappropriate Parzen window size used. Figure B.1, below, shows how the maximum value obtained for the correntropy function is close to 47 which is much lower than the optimal value of 49 (obtained using a $\sigma=0.5$).

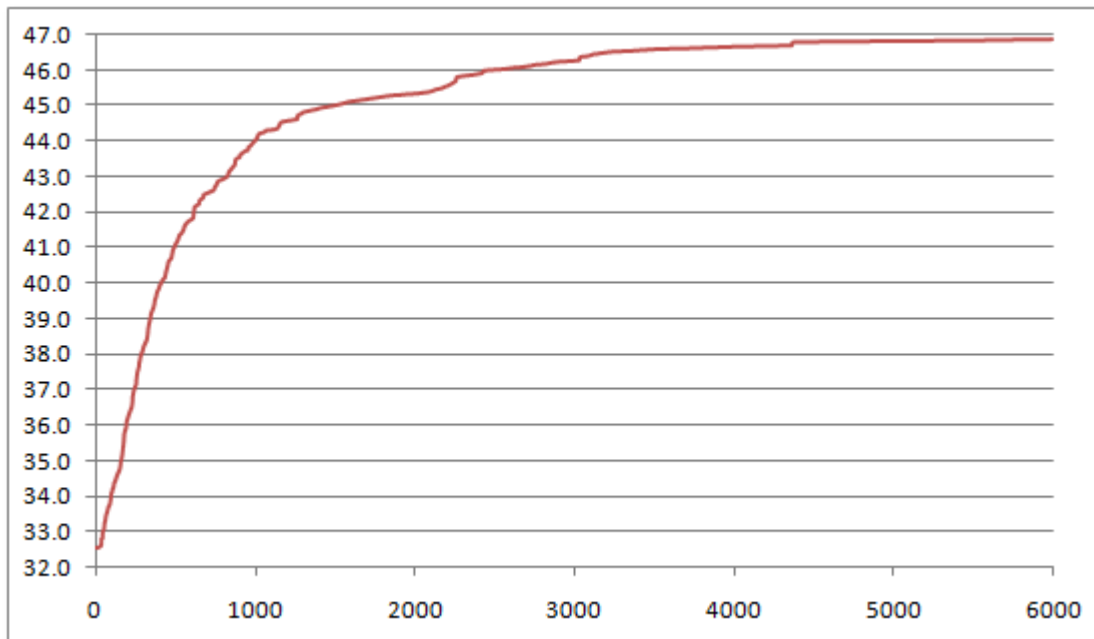


Figure B.1 - Line 6 with inverted error: correntropy function evolution using $\sigma=0.2$.

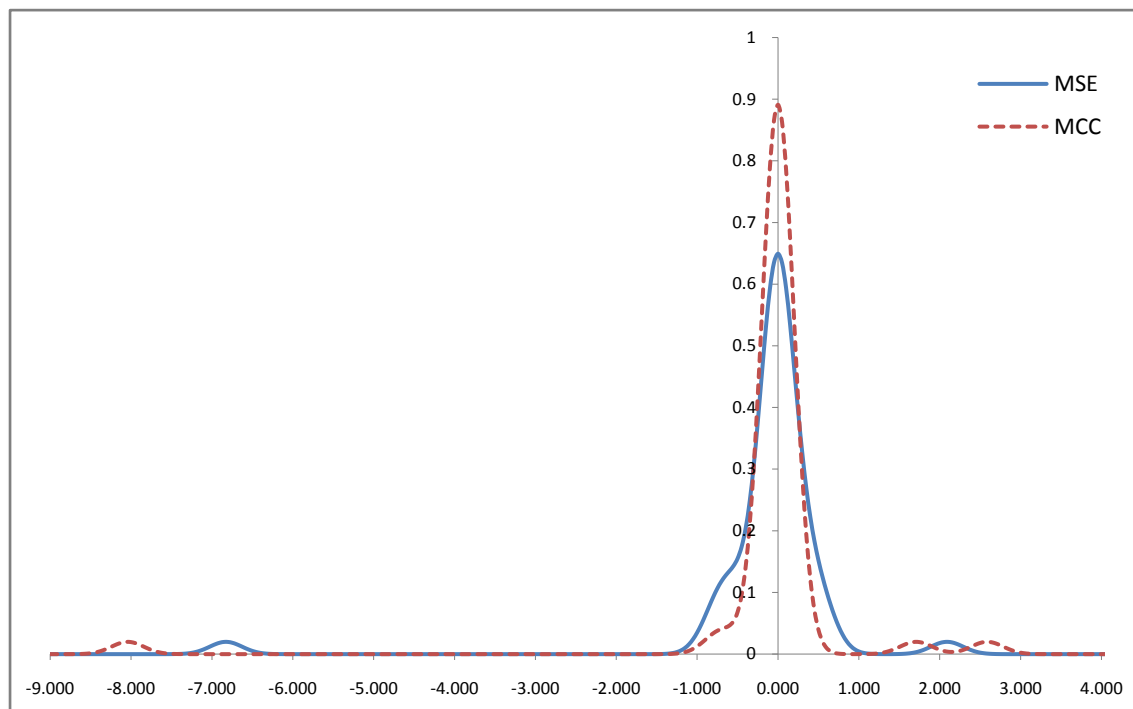


Figure B.2 - Line 6 with inverted error not converging: Gaussian error distribution using a Parzen window scaling parameter $\sigma'=0.2$.

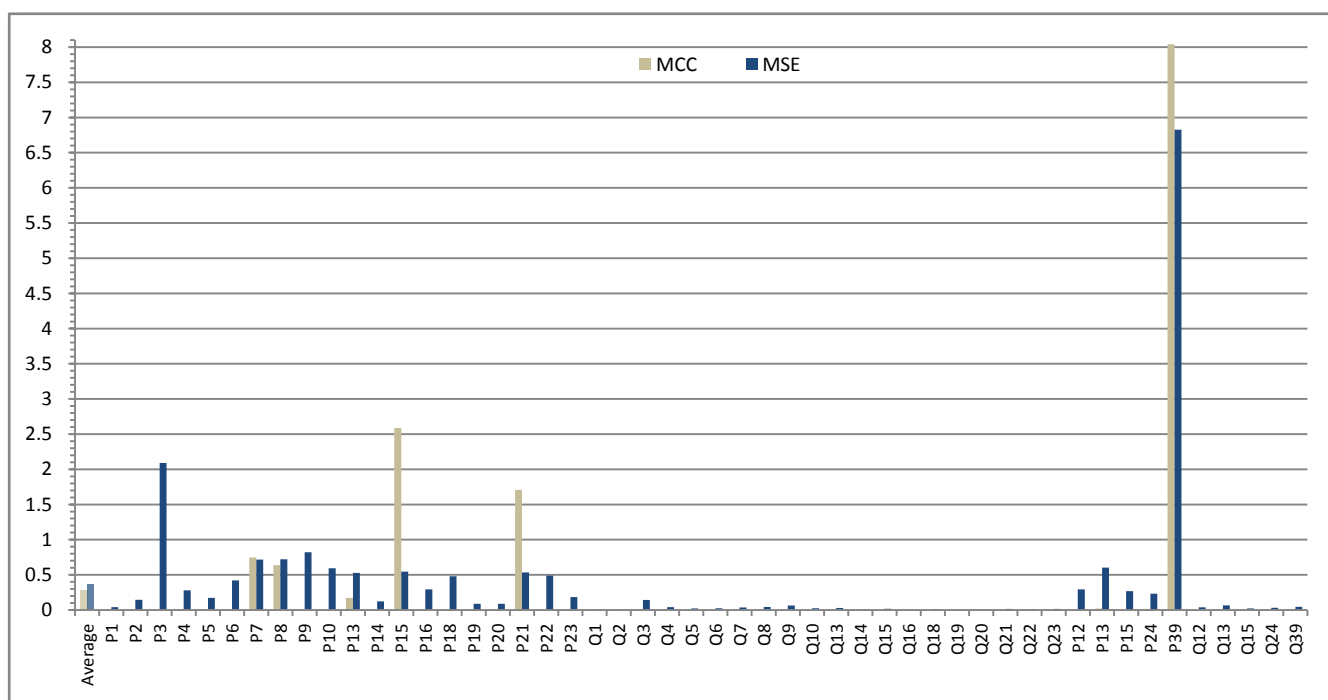


Figure B.3 - Line 6 with inverted error not converging: Gaussian error distribution using a Parzen window size $\sigma=0.2$.

A close inspection on figure B.3 shows how the 0.2 Parzen window size wasn't enough to include all the "small" errors and isolate only the gross error. Moreover, we see 2 additional errors (P_{15} and P_{21}) being detected as outliers as well, thus this can't be considered a feasible solution. That way, only by increasing the Parzen window size, one is able to obtain an acceptable solution.

Annex C

EPSO Algorithm

To solve mathematically complex optimization problems, metaheuristics (also known as black-box) such as evolutionary algorithms are being increasingly used due to their inherent advantages. In order to locate or find (one or more local) optimum solutions of a given problem in the context of a domain name search algorithm, two key mechanisms are inevitably considered. The first is movement generator that's able to produce new feasible solutions and the second is an evaluation procedure to quantify the alternatives in the search space, thus classifying them according to the optimization objectives.

It is important to have smart strategies for searching in the solution space. The following three “intelligent” strategies are the Genetic Algorithms (GA), Particle Swarm Optimization (PSO) and finally the used algorithm in this thesis in Chapter 5, the Evolutionary Particle Swarm Optimization (EPSO), which combines the best qualities of the GA and PSO algorithms.

In the following two sub-chapters, a brief description of the GA and PSO algorithms will be made in order to elucidate the reader on the advantages of the more recent approach using the EPSO.

C.1 Genetic Algorithms

Genetic Algorithms are implemented by creating an initial population (with random values, but, whenever possible, optimized previously to obtain faster and better convergence) upon which the evolutionary mechanisms are applied. The most common used genetic algorithm form was described by Goldberg in [48]. Each individual of the generated population represents a solution for the problem. These evolutionary mechanisms, like the majority of meta-heuristics were inspired on biological species evolution¹⁴ like the survival of the fittest and the inter-individual communication to obtain better results for the group.

The main evolutionary steps applied to the population are:

- a) Selection
- b) Mutation
- c) Recombination

The selection is based on the competition among alternatives to identify the best, which will become seeds of the next generation. As a result of the evaluation of each solution (individual), two types of selection techniques can be used: elitist and stochastic. The first

¹⁴ Such as inheritance, mutation, selection, and recombination.

one guarantees that only the best solutions survive to the next generation. The stochastic “tournament” uses a (usually small) probability that an unfit individual is able to survive. The latter usually gives best results because even the “bad” individuals carry important genes that can help create diversity, thus a better individual in the future like what happens in the species evolution.

Mutation is a unary operator, acting on a single individual and responsible for generating a new solution (new individual), by applying random modifications to it. This is a procedure used to ensure higher diversity in solutions, thus helping the algorithm to avoid excessively similar solutions, or in other words, lack of diversity which helps avoiding local optimums.

Recombination (also referred as crossover) generates new individuals (new points in the search space) by randomly mixing characteristics from more than one parent (solutions previously found).

Figure C.1, below, illustrates how the Genetic Algorithm operation procedure works, including the previously detailed procedures: Selection, Mutation and Recombination (crossover). In a first step, a randomly initial population is generated. Then, all the individuals of the initial population are evaluated. After the evaluation procedure, if the convergence test is positive, the best individual is presented as the final solution, otherwise it proceeds to the selection, recombination and mutation mechanisms, returning then to the evaluation procedure, until the convergence test is positive or the maximum number of generations (iterations) is reached.

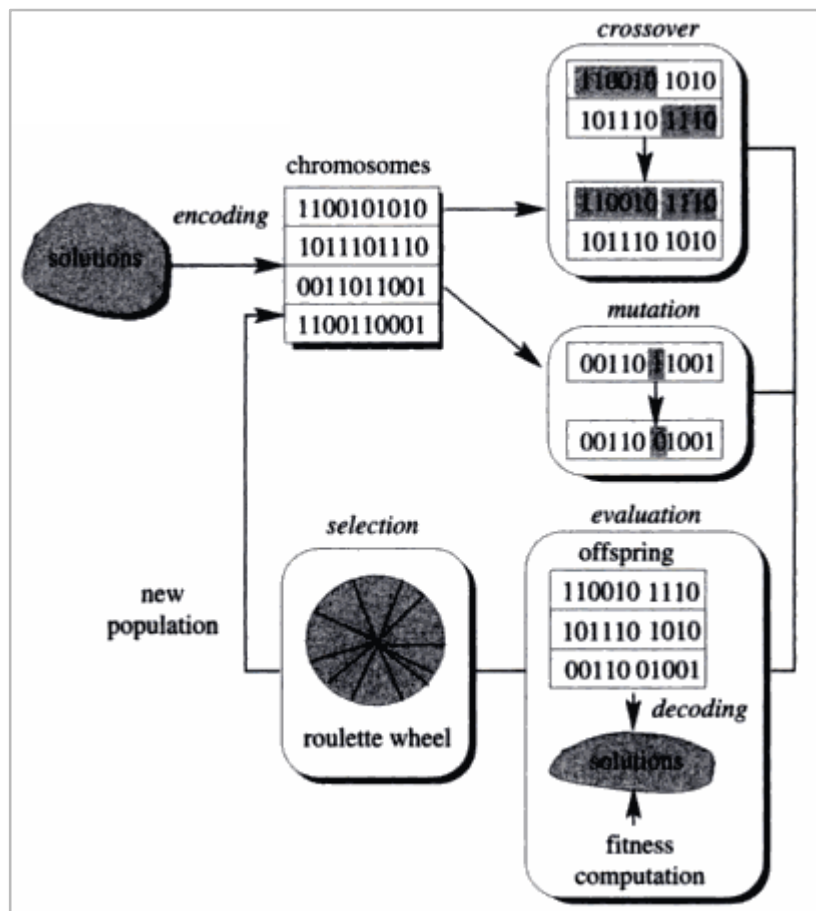


Figure C.1 - Genetic algorithm diagram scheme [49].

C.2 Particle Swarm Optimization (PSO)

The Particle Swarm Optimization (PSO) is a relatively new algorithm, having been introduced by James Kennedy and R. C. Eberhardt in 1995 in reference [50], which reveals competitive characteristics when compared with older meta-heuristic algorithms.

The main idea present in this algorithm is the social behaviour of living organisms such as swarms of bees, birds or schools of fish, in a way that the group (swarm) and each individual “respond” to the movement as whole.

Each individual (solution) is represented as a particle that will move in the domain search space affected by the influence of three vectors:

1. Inertia
2. Memory
3. Cooperation

The first vector - **Inertia** - will force the particle in the same direction that it was going. This is equivalent to the physical movement inertia that makes it impossible for an object to instantaneously vary its movement speed and direction.

The second vector - **Memory** - will “attract” the particle towards the best position achieved so far by it.

The third vector - **Cooperation** - will “attract” the particle towards the best global position archived by the swarm, regardless what particle achieved it.

Each of the three components will be multiplied by weights that determine the influence of each vector in the definition of the new position taken by the particle.

The movement equations are thus given by:

$$v_i(t+1) = w_{i,0} \times v_i(t) + R(w_{i,1}) \times (p_i(t) - x_i(t)) + R(w_{i,2}) \times (g(t) - x_i(t)) \quad (C.1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (C.2)$$

Where,

$v_i(t)$ - Velocity on instant t for particle i;

$x_i(t)$ - Position on instant t for particle i;

$p(t)$ - Particle best position on instant t;

$g(t)$ - Global best position on instant t;

$w_{i,0}$ - Inertia coefficient;

$w_{i,1}$ - Memory coefficient;

$w_{i,2}$ - Cooperation coefficient;

R - Random distribution of the weights.

By observing equation (C.1) one concludes that the inertia term is composed by $w \times v(t)$, the memory term is composed by $R(w) \times (p(t) - x(t))$ and the cooperation term is composed by $R(w) \times (g(t) - x(t))$.

In PSO, unlike other evolutionary algorithms, there is not any competition among particles or self adaptation of their characteristics. In reality, if the cooperation term was removed, each particle would completely ignore the others when defining her new position.

Thus, the particle's new position is defined by the sum of the three previously mentioned vectors, and is illustrated in Figure C.2:

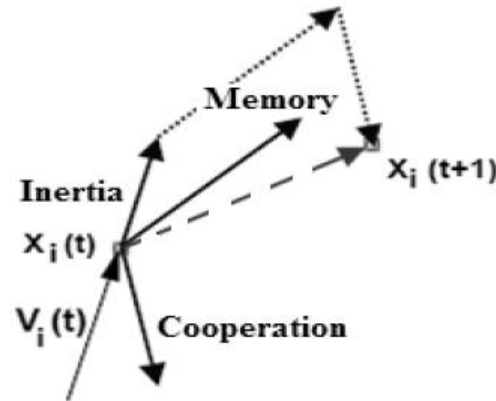


Figure C.2 - New particle position: PSO vector diagram.

The operation procedure of the PSO algorithm is schematized in Figure C.3, below. By analysing it, one observes that the first step is to generate a randomly initial population. Next step involves the evaluation of the population according to the specified criteria. After the evaluation procedure, if the convergence test is positive, the best individual in the swarm is presented as the final solution, otherwise the particles are moved before being evaluated again. These last 3 steps will repeat themselves until the convergence test is positive or the maximum number of iterations is reached.

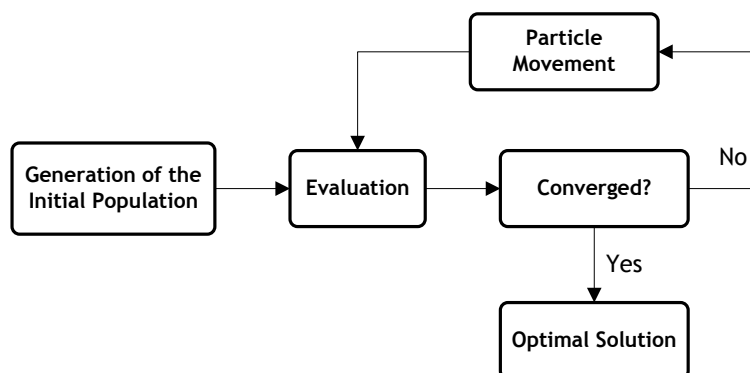


Figure C.3 - Particle Swarm Optimization algorithm diagram.

C.3 Evolutionary Particle Swarm Optimization (EPSO)

As previously verified, the main disadvantage of the PSO is that it's not a self-adaptive algorithm. Some of the used parameters need to be externally defined by the user, which determines "fixed weights" for the inertia, memory and cooperation terms in the process. Thus, the movement mechanism won't evolve learning from the experience in previous iterations, in other words, this search method is less efficient.

To address the disadvantage of the PSO, the Evolutionary Particle Swarm Optimization (EPSO) was developed which is a hybrid optimization algorithm, developed at INESC Porto [46; 47]. As previously said, the EPSO gathers the best qualities of the Genetic Algorithm and of the Particle Swarm Optimization.

One important difference between the PSO and EPSO algorithms, include the adoption of a stochastic star communication topology, instead of the deterministic scheme usually adopted in PSO. This has the advantage of sharing the knowledge of each particle's knowledge of the global best position, controlled by a communication probability p , which is self-adaptive throughout the algorithm run and also externally defined. The stochastic star communication topology means that a particle will ignore the global best knowledge on some iterations and include it in other iterations. This not only allows a more local search by each particle, but also allows the elimination of noise, by allowing the dynamics of particle movement to be more stable and avoiding premature convergence.

The other main differences between EPSO and PSO are the selection, replication, weight mutation and movement (reproduction) procedures, as can be seen comparing Figures C.3 and C.4.

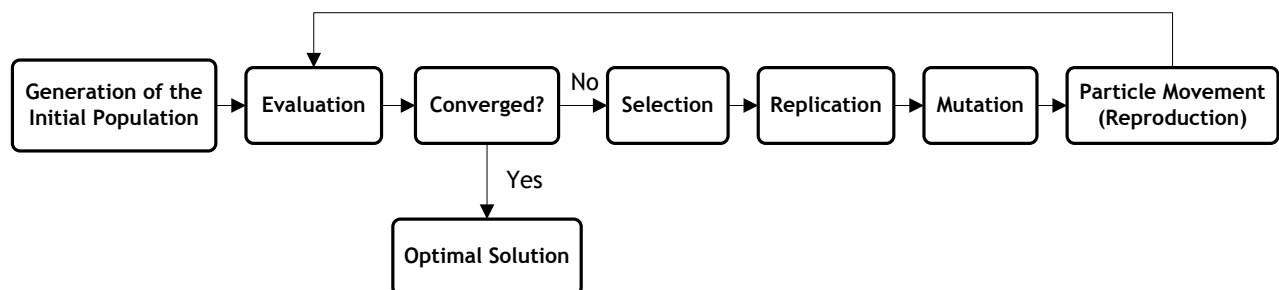


Figure C.4 - Evolutionary Particle Swarm Optimization algorithm diagram.

The selection procedure is what makes this algorithm actually evolutionary, by forcing competition between particles. This way, only the best particles (evaluated previously) "survive" into the next generation (iteration).

The replication procedure is used to create copies of every particle, which will be mutated afterwards, obtaining this way more diversity in the population, which helps avoiding local optimums.

The mutation procedure is one of the key improvements made by the EPSO over PSO. Here, the weights from the previous iteration (w) are mutated so that "new" weights (w^*) are calculated for the next iteration.

The new weights are obtained by multiplying the weight w with random number from lognormal distributions:

$$w_i^* = w_i \left[\log N(0,1) \right]^\tau \quad (C.3)$$

or by random number Gaussian distributions:

$$w_i^* = w_i + \sigma N(0,1) \quad (C.4)$$

The τ and σ learning (mutation) parameters are externally defined by the user.

The most efficient EPSO applications mutate, not only the weights affecting the components of movement, but also randomly disturb the global best position giving:

$$g^*(t) = g(t) + w_{i,3}^* \times N(0,1) \quad (C.5)$$

where $w_{i,3}^*$, determines the size of the search space around $g(t)$ where is most likely to find the real best global solution. This helps particles to avoid getting “stuck” in local optimums, in cases where the cooperation term evolves to become dominating while the other terms fade out.

The EPSO movement equations are then given by:

$$v_i(t+1) = w_{i,0}^* \times v_i(t) + w_{i,1}^* \times (p_i(t) - x_i(t)) + w_{i,2}^* \times P \times (g^*(t) - x_i(t)) \quad (C.6)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (C.7)$$

Where,

$v_i(t)$ - Velocity on generation t for particle i ;

$x_i(t)$ - Position on generation t for particle i ;

$p_i(t)$ - Best point found by particle i in its past life;

$g^*(t)$ - Best overall point found by the swarm;

$w_{i,0}$ - Inertia weight coefficient;

$w_{i,1}$ - Memory weight coefficient;

$w_{i,2}$ - Cooperation weight coefficient;

P - Diagonal matrix with each element being a binary variable equal to 1 with a given communication probability p and 0 with probability $(1-p)$.

Note: the $*$ symbol indicates that the weights were mutated.

Thus, the particle's new position is defined by the vector sum represented in Figure C.5:

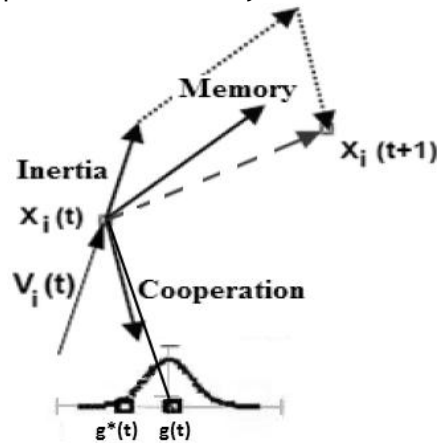


Figure C.5 - New particle position: EPSO vector diagram.

Annex D

Analysis of power system security

The energy management system (EMS) and distribution management system (DMS) are the ones responsible for the control, supervision and implementation of the economy-security functions of the electric utility companies.

The concept of network security is linked to the probability of maintaining an adequate supply of electric power, without decisively violating the established technical and operation constraints. Therefore, the higher a security level is, the lower its probability of loss of load, and vice-versa. However, higher security levels come at higher investment costs and it's the SO's role to balance both. Thus, to avoid damaging equipment and blackouts, security oriented control actions are implemented.

A loss of transmission equipment or generation units is called a contingency. Therefore, a Contingency Analysis (CA) is done to evaluate the possible contingencies that could occur and detect the potential damaging ones. These potential damaging contingencies are those that would lead the system into a state of emergency, thus need to be examined in detail.

It's known that the management of the "security-constrained optimal control of an electric power system generation-transmission" [51] is very difficult. Moreover, the difficulty is increased as the systems get bigger and more complex with interconnected local and regional area networks.

An example of a huge interconnected electric network system is the one in Europe by the UCTE¹⁵ TSO members, where Portugal is included. This kind of association has many advantages, including no need to have a large primary reserve available at all times, because the entire UCTE network is capable to supplying the remaining power if needed, to a certain degree. This also keeps the system's frequency much more stable than if it was an isolated electric network, such as what happens, for example, in islands like *Madeira* and *Açores* where an isolated network is forcibly used, thus being much more vulnerable to faults like short-circuits or loss of generation power in maintaining the frequency values within the specified values.

On the other hand, there are also some disadvantages like, for example, if one (or more) of the UCTE members is unable to control and correct its contingencies¹⁶ in due time, there's a danger of contaminating the remaining network. The most serious situation (on the number of TSO's involved and amplitude of frequency deviation) in the history of the UCTE was the

¹⁵ UCTE - Union for the Co-ordination of Transmission of Electricity; "coordinates the operation and development of the electricity transmission grid from Portugal to Poland and from the Netherlands to Romania and Greece. UCTE, the association of transmission system operators in continental Europe in 24 countries, provides a reliable market platform to all participants of the Internal Electricity Market and beyond" [52]; responsible for supplying energy to about 450 million people.

¹⁶ In power systems a contingency is defined as a loss in generation units or transmission equipment or even natural disasters such as thunderstorms.

incident of 4th of November 2006. A brief description of the incident is shown below and further information can be found in references [53; 54; 55; 56; 57].

The power consumption, at around 22h, in the control area of *E.ON Netz* was approximately 13500 MW and there was an injected wind power of 3300 MW at the same time. Due to the transits, the lines towards West were already loaded, being this a normal operating situation.

The following is a quotation of the press release article [55] a few days after the incident:

“At 21:38h, both circuits of the 380-kV-line Conneforde-Diele were switched off in order to secure the passing of the Ems river by a ship. A routine simulation of the switching-off of the a.m. line was computed in advance and did not bring up concerns about this switching maneuver. Following the switching-off of the said line, the energy flow was transferred to other lines in the South. This situation was still stable.

After some time, a higher load appeared on the E.ON Netz grid, especially on the line Wehrendorf-Landesbergen (East-Westphalia) between the transmission systems of RWE TSO and E.ON Netz.

(...)

At 22:10h, the lines Wehrendorf-Landesbergen and Bechterdissen-Elsen tripped. Because of the tripping of the 2 a.m. high-voltage lines, further tie-lines were overloaded within seconds and tripped in cascade. This led to a split of the European UCTE interconnected network into 3 islands with different frequencies (see our yesterday information [53]). Since at this moment, the geographic generation power was no longer evenly spread over the system, the North-Eastern part of the system (after the split) had a generation surplus of approx. 6000 MW. It appears from today's available data that both the Western and South-Eastern parts had on the contrary a lack of power. In order to re-establish the balance between generation and consumption, load-shedding (cut of power supply to industry and household customers) was performed as follows:

Table D.1 - UCTE incident: amount of cut loads by the TSO's

Country/TSO	Load shed (MW)
Austria/APG AG	1500
Austria/Tiwag Netz	40
Belgium/Elia	800
France/RTE	5200
Germany/E.ON Netz	400
Germany/RWE TSO	2000
Italy/TERNA	1500
Netherlands/TenneT	400
Portugal/REN	500
Spain/REE	2100
Slovenia/ELES	100

Upon action of the transmission system operators, the European UCTE network was reconnected after 38 minutes and the customers were progressively supplied again.” [55]

Following an investigation to determine the causes of this incident, the report [57] had found two root causes and five critical factors.

Root causes:

- Non fulfilment of the N-1 Criterion¹⁷;
- Inappropriate inter-TSO coordination during this event;

Critical factors:

- Generators related issues¹⁸; TSO access to real time information on generation;
- Range of action available to dispatchers for handling grid congestions;
- TSO/DSO (Distribution System Operators) co-ordination in the context of defence and restoration plans;
- Co-ordination of resynchronization procedures during the event;
- Enhancement of training of dispatchers.

Even though a generalized blackout of the European network was avoided by the fast reaction of the TSO's and the protection systems installed, this incident is a fine example why system security analysis is very important for the electric systems' daily operation. It is also one area that is constantly evolving as new technological systems are developed. A grave incident like this does have a positive side, since it allows the TSO's and respective regulators to learn from the mistakes and take measures to prevent it from happening again, thus effectively increasing the system's security in the future. Furthermore, the lessons learned can also be used by the other TSO's in analogous associations in different continents, like America or Africa to better secure their system.

The fundamentals of the methodology in power system security analysis were first introduced by Dy Liacco in reference [58], also considering some generalizations made to the concept of security control in industrial systems by Sollberger [59]. This came as an important step in the advance of system security comprehension by electric utility companies.

Figure D.1 represents this methodology, represented by the 3 operating states.

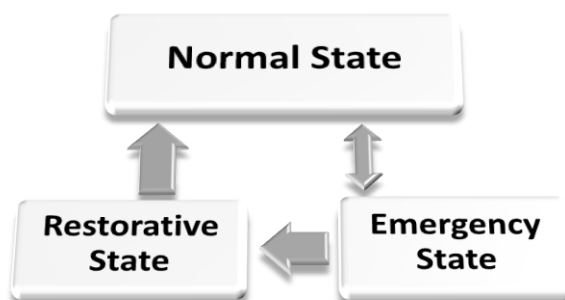


Figure D.1 - Power system security: state diagram of system operation [14]. The gray arrows represent contingencies in the electric system.

¹⁷ The N-1 security rule requires that a single incident should not put at risk the secure operation of the interconnected network. Such incidents are, for example, tripping of a generation unit, a transmission line or transformer. In particular, the N-1 principle aims at avoiding cascading effects.

¹⁸ Mainly from the wind power and CHP generators that were more sensible to the frequency variation.

A system is said to be in a “normal operating state” if all the loads are being supplied by the existing systems power generation without violating any operating limits (e.g. node voltage limits, line limits, transformer limits, generator limits, etc.) and all constraints are satisfied.

When severe disturbances occur (for e.g. short circuits, big load change and/or loss of generation), the system may jump to a new normal state or could go to either an emergency or a restorative operating state. The system goes to the “emergency state”, if the operating constraints were not met. However, it goes to the “restorative state” if the operating constraints are satisfied but the load constraints are not.

One can divide the referred emergency state in two type causes:

- i. The first is called a “steady-state emergency” and occurs when, after a disturbance, the power system continues operating in a stable way, although the operating constraints are not within the established limits (i.e., exceeding the limits of any equipment or anomalous voltage levels). This type of emergency condition might be tolerated for a reasonable period of time, which allows for corrective measures to be executed. These corrective measures need to be as effective as possible, to reduce or prevent damaging the overloaded equipment;
- ii. The second is called a “dynamic instability” and occurs when, because of a disturbance (e.g. short circuit), the power system becomes unstable, meaning that the operating and load constraints are not being respected. This type of emergency condition has transient characteristics, thus occurring in a very short period of time and, to avoid going to the restorative state (meaning a partial or complete blackout), fast and accurate corrective measures need to be executed.

To summarize, one can say that the main objective of security control is to maintain the power system in the normal operation state, effectively preventing or minimizing the transition from the normal operating state into either the emergency or the restorative state.

Years later, Brian Stott, Ongun Alsac, and Alcir Monticelli introduced a paper [51] that presents an overview of the power system security problem, and also some methodologies and formulations to deal with the challenges of optimal power flow algorithms, having the economy-security¹⁹ exploration control pair always present, as well as the system’s contingency constraints that need to be satisfied. It is also discussed the need for the preventive control actions by simulation, executed by the contingency analysis module of the EMS/DMS.

Additionally, the operating states’ methodology introduced by Dy Liacco in [58] (represented in Figure D.1) was further improved in [17; 51], that introduces a 6 level power system security classification, as illustrated in Figure D.2, below.

¹⁹ The main goal of the economy-security control is to explore the power system at the lowest costs possible, although still assuring its survival when subjected to the emergency conditions, or even avoid them completely. This implies that the system operators (TSO, DSO ...) have to operate the system as close to the security limits as possible, in order to delay investment.

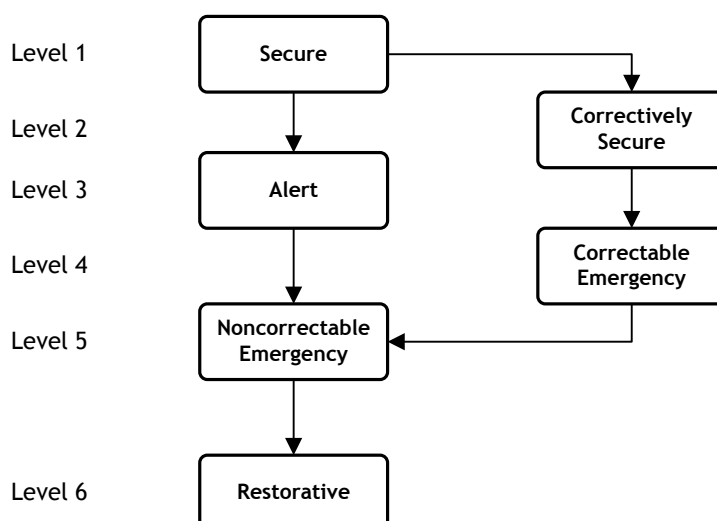


Figure D.2 - Power system security: extended 6 level diagram [17; 51]. The arrows represent the involuntary level transitions caused by contingencies.

As can be seen in Figure D.2, the six security level states that could occur, define the sequence of control actions that should be taken in consideration during a corrective or preventive procedure. Hence, a brief description of each security level is presented below:

- *Level one - Secure:* in this level, the system supplies all loads without violating any of the operating limits. Furthermore, the system will remain operating within the established limits even if one contingency arises. Therefore, when in this level and after a contingency occurs, the system will not go to the emergency state. As a result no corrective measures will have to be taken;
- *Level two - Correctively Secure:* like in level one, all loads are supplied without violating the operating limits. Nevertheless, to avoid loss of load, control actions (e.g. active power control of the generators - primary reserve) need to be quickly performed in order to make the necessary corrections on the violated operating limits caused by a contingency. Compared to the previous level, this is a more economic level because it allows, for example, a smaller spinning reserve²⁰, thus reducing the “redundancy” costs. However, it needs post-contingency measures to be performed by the DMS/EMS. By running an optimal power flow with security constraints and including pos-contingency rescheduling, it is possible to identify the adequate control actions prior to the contingency occurrence. It’s then clear that this level has a greater risk of ending up in the emergency state than level one;
- *Level three - Alert:* a system at this level, like in level one and two, has all loads supplied without violating its operating limits. Yet, some load has to be cut, in order to fix some of the exceeded limits caused by contingencies. The system can be brought back to levels one or two by preventive rescheduling using an optimal power flow (OPF) with security contingency constraints;
- *Level four - Correctable Emergency:* in this level all loads are supplied, but some operating limits are violated, although these can be corrected without loss of load. By

²⁰ Spinning reserve is defined as the extra generating capacity that’s available by increasing the output power of generators already connected to the power system. Most generators can achieve this, by increasing the torque applied to the turbine’s rotor.

performing corrective actions, the system can be brought back to level three. Nevertheless, only the medium-long term limits used in levels 1-3 can be violated and not the short-term ones;

- *Level five - Noncorrectable Emergency*: if the preventive measures are not effective or fast enough, level five is reached. Like in level four, all loads are supplied, but some operating limits are nevertheless violated, like level three, these violations can't be corrected without loss of load. The amount and location of the load loss can be optimized using an OPF routine. It should be noted that, this load loss optimization is very important since some loads have higher priorities than others (e.g. paper industry, hospitals, government agencies, etc. are much more vital than street illumination, hydro pumping ...);
- *Level six - Restorative*: in this level, a system has no violations in the operating limits but loss of load has occurred. The restorative control, in this level, tries to return the system back to levels one or two. Restorative procedure takes place, for example, when a full or partial blackout occurs.

The current system's security level determination and the suitable control action to be taken at each level are carried out by the various EMS/DMS economy-security functions.