Multilevel MATE Algorithm for the Simulation of Power System Transients with the OVNI Simulator

by

Mažana Lukić Armstrong

B.Sc., Fakultet Elektrotehnike i Računarstva, Zagreb, Croatia, 1995
M.A.Sc., The University of British Columbia, Vancouver, Canada, 2001

A THESIS SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

The Faculty of Graduate Studies

(Electrical and Computer Engineering)

The University Of British Columbia
December, 2006
© Mažana Lukić Armstrong 2006
Abstract

The existing MATE ("Multi Area Thévenin Equivalent") concept implemented in the real-time simulator OVNI provides a means of partitioning large systems of equations into subsystems connected through links. The subsystems are solved independently, with the overall solution integrated at the level of the links. MATE partitioning enables fast simulation of power system networks in two distinct ways: first, it allows parallel processing in a multi-machine environment and second, it allows integration of different solution techniques for individual subsystems.

In this thesis, we generalize the concept of MATE with the new Multilevel MATE concept, in which each subsystem becomes the basis for a new level of MATE partitioning. The new concept allows subsystems to be further partitioned into subsubsystems that, for example, are of a constant and changing nature. Partitioning at the subsystem level leads to higher overall solution efficiency. Furthermore, power system components can be described using Multilevel MATE by their nodal and/or branch equations on the subsystem level in OVNI. A three-phase induction machine model is an example of a power system component that is naturally described with branch equations. Multilevel MATE also provides a convenient framework for the incorporation of controllers as well as nonlinear elements.

The software implementation of OVNI written in this work is based on the Multilevel MATE algorithm. Models of power system components have been implemented and tested using the newly developed concept. A test case describing a doubly-fed induction generator wind turbine system has been modelled and studied as a practical example of the new solution scheme's capabilities.
# Table of Contents

Abstract ................................................................. ii

Table of Contents ...................................................... iii

List of Tables ........................................................... vii

List of Figures .......................................................... viii

Acknowledgements ...................................................... xiii

1 Introduction .......................................................... 1
   1.1 Background ...................................................... 1
   1.2 Motivation and Objectives ...................................... 2
   1.3 Thesis Organization ........................................... 3
   1.4 Thesis Contributions .......................................... 4

2 An Overview of Power System Simulation Tools and Techniques ..... 6
   2.1 Introduction .................................................... 6
   2.2 Techniques for Simulation of Large Power System Networks .... 7
       2.2.1 Sparsity .................................................. 8
       2.2.2 Diakoptics .............................................. 10
       2.2.3 Modified Nodal Analysis .............................. 10
   2.3 Summary ....................................................... 12

3 Multilevel MATE ...................................................... 13
   3.1 Introduction ................................................... 13
   3.2 MATE System Partitioning ................................... 14
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.3</td>
<td>Multilevel MATE Concept</td>
<td>16</td>
</tr>
<tr>
<td>3.3.1</td>
<td>Example: An Ideal Switch Implementation with Multilevel MATE</td>
<td>20</td>
</tr>
<tr>
<td>3.4</td>
<td>Computational Efficiency of Multilevel MATE Partitioning</td>
<td>22</td>
</tr>
<tr>
<td>3.5</td>
<td>Commentary on Computational Efficiency of the Multilevel MATE Partitioning</td>
<td>25</td>
</tr>
<tr>
<td>3.6</td>
<td>Solution of Nonlinearities within the Multilevel MATE Concept</td>
<td>25</td>
</tr>
<tr>
<td>3.6.1</td>
<td>Example: A Nonlinear Control Function Implementation with Multilevel MATE</td>
<td>28</td>
</tr>
<tr>
<td>3.7</td>
<td>Multilevel MATE Block Diagram</td>
<td>31</td>
</tr>
<tr>
<td>3.8</td>
<td>Summary</td>
<td>32</td>
</tr>
<tr>
<td>4</td>
<td>Modelling with Multilevel MATE</td>
<td>34</td>
</tr>
<tr>
<td>4.1</td>
<td>Introduction</td>
<td>34</td>
</tr>
<tr>
<td>4.2</td>
<td>Modelling of Ideal Voltage Sources</td>
<td>35</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Commentary on Modelling of Ideal Voltage Sources</td>
<td>36</td>
</tr>
<tr>
<td>4.3</td>
<td>Modelling of Ungrounded Circuits and Subsystems</td>
<td>36</td>
</tr>
<tr>
<td>4.3.1</td>
<td>Ungrounded Circuits</td>
<td>37</td>
</tr>
<tr>
<td>4.3.2</td>
<td>Ungrounded Subsystems</td>
<td>42</td>
</tr>
<tr>
<td>4.4</td>
<td>Modelling of Electric Machines</td>
<td>46</td>
</tr>
<tr>
<td>4.4.1</td>
<td>Phase-Domain Induction Machine Model</td>
<td>47</td>
</tr>
<tr>
<td>4.4.2</td>
<td>Phase-Domain Synchronous Machine Model</td>
<td>64</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Commentary on Modelling of Electric Machines</td>
<td>68</td>
</tr>
<tr>
<td>4.5</td>
<td>Modelling of Controllers</td>
<td>68</td>
</tr>
<tr>
<td>4.5.1</td>
<td>Example: Modelling of a Single-Machine Infinite-Bus System in $dq$ Coordinates with Multilevel MATE</td>
<td>70</td>
</tr>
<tr>
<td>4.5.2</td>
<td>Commentary on the Simultaneous Solution of Controller Equations</td>
<td>74</td>
</tr>
<tr>
<td>4.6</td>
<td>Summary</td>
<td>77</td>
</tr>
<tr>
<td>5</td>
<td>Modelling and Simulation of a DFIG Wind Turbine System in OVNI</td>
<td>78</td>
</tr>
<tr>
<td>5.1</td>
<td>Introduction</td>
<td>78</td>
</tr>
<tr>
<td>5.2</td>
<td>Wind Turbine Generators</td>
<td>79</td>
</tr>
</tbody>
</table>
Table of Contents

5.3  Doubly Fed Induction Generator Wind Turbines .......................... 80
5.4  DFIG Model Structure ......................................................... 80
5.5  Doubly Fed Induction Generator Model ..................................... 81
      5.5.1 Two-mass Representation of a DFIG WTG Shaft ...................... 81
5.6  Voltage Converter Model and Control ....................................... 83
      5.6.1 Stator-side Converter Model ........................................ 85
      5.6.2 Stator-side Converter Control ...................................... 88
      5.6.3 Rotor-side Converter Model ........................................ 89
      5.6.4 Rotor-side Converter Control ...................................... 92
5.7  Wind Turbine Model ............................................................ 94
5.8  Maximum Power Tracking ....................................................... 95
5.9  Implementation of a DFIG Wind Turbine System in OVNI ................. 97
5.10 DFIG Wind Turbine Test System ............................................ 98
5.11 Simulation Results ............................................................. 99
      5.11.1 Transient Response of the DFIG Wind Turbine to a Step Decrease in Wind Velocity .................................................. 99
      5.11.2 Transient Response of the DFIG Wind Turbine to a Three-phase Short Circuit .................................................. 107
      5.11.3 Commentary on the Comparison of Transient Simulation Results Obtained with EMTP-type and Stability-type Tools .................................................. 108
5.12 Summary ................................................................. 108

6  Conclusion .............................................................................. 119
6.1  Summary of Contributions ................................................... 119
6.2  Recommendation for Future Research ..................................... 120

Bibliography ................................................................. 122

Appendices ................................................................. 128

A  Trapezoidal Integration Rule ............................................. 129

B  Reference Frame Transformations ..................................... 131
Table of Contents

B.1 Clarke Transformation .................................................. 131
B.2 $dq$ Transformation ...................................................... 132

C Digital Pass-Band Filter for Filtering Stator-Flux ................... 134

D Test System Parameters .................................................... 137
  D.1 Parameters of the 2250 hp Induction Motor from Chapter 4.4.1 ... 137
  D.2 Parameters of the 835 MVA Synchronous Generator from Chapter 4.4.2 138
  D.3 Parameters of the 7.5 kW Doubly Fed Induction Generator from Chap­
      ter 5.10 ................................................................. 139
### List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Partitioning of subsystems</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>Iterations for the nonlinear controller equation</td>
<td>31</td>
</tr>
<tr>
<td>D.1</td>
<td>2250 hp induction motor parameters</td>
<td>137</td>
</tr>
<tr>
<td>D.2</td>
<td>835 MVA steam turbine generator parameters</td>
<td>138</td>
</tr>
<tr>
<td>D.3</td>
<td>7.5 kW doubly fed induction generator wind turbine system parameters</td>
<td>139</td>
</tr>
</tbody>
</table>
## List of Figures

2.1 Independent voltage source ............................................. 11

3.1 An example of a system that demonstrates the MATE concept .... 14
3.2 General description of a link ........................................... 15
3.3 An example of a system that demonstrates the Multilevel MATE concept 16
3.4 An example system to demonstrate ideal switching using the Multilevel MATE concept .................................................. 20
3.5 Multilevel MATE vs. single-level MATE computation speedup .... 24
3.6 An example system with a nonlinear control function ............ 29
3.7 Flow chart of iteration process for nonlinear controllers ........ 32
3.8 Block diagram of Multilevel MATE implementation ............... 33

4.1 An example system to demonstrate modelling of ideal voltage sources 35
4.2 An example system to demonstrate modelling of ungrounded circuits 37
4.3 Modelling of an ungrounded transformer circuit with the Multilevel MATE approach ..................................................... 38
4.4 Introducing a common reference point to the ungrounded secondary side of an ideal transformer ....................................... 39
4.5 Introducing a common reference point to the ungrounded network with two ungrounded ideal voltage sources .................... 40
4.6 Introducing a common reference point to the ungrounded network with two grounded ideal voltage sources ..................... 42
4.7 Grounded system with an ungrounded subsystem .................... 43
4.8 Rotor and stator circuits of an induction machine .................. 48
4.9 Coupled inductors .......................................................... 49
4.10 Discrete phase domain induction machine electrical model ....... 51
List of Figures

4.11 Electrical circuit configuration for testing the phase-domain induction machine model implemented with Multilevel MATE ........................ 54
4.12 Stator phase current during free acceleration of a 2250 hp induction motor ........................................ 59
4.13 Rotor phase current during free acceleration of a 2250 hp induction motor ........................................ 59
4.14 Electromagnetic torque during startup of a 2250 hp induction motor .................................................. 60
4.15 Rotor speed during startup of a 2250 hp induction motor ................................................................. 60
4.16 Torque-speed characteristics during startup of a 2250 hp induction motor ......................................... 61
4.17 Torque-speed characteristics during step changes in load torque of a 2250 hp induction motor ................. 61
4.18 Stator phase current during step changes in load torque of a 2250 hp induction motor .................................. 62
4.19 Rotor phase current during step changes in load torque of a 2250 hp induction motor .......................... 62
4.20 Electromagnetic torque during step changes in load torque of a 2250 hp induction motor ....................... 63
4.21 Rotor speed during step changes in load torque of a 2250 hp induction motor ..................................... 63
4.22 Rotor and stator circuits of a synchronous machine ........................................................................... 64
4.23 Stator phase current during a step increase in input torque of a 835 MVA steam turbine generator .................. 65
4.24 Field current during a step increase in input torque of a 835 MVA steam turbine generator ...................... 65
4.25 Electromagnetic torque during a step increase in input torque of a 835 MVA steam turbine generator .................. 66
4.26 Rotor speed during a step increase in input torque of a 835 MVA steam turbine generator ....................... 66
4.27 Rotor angle during a step increase in input torque of a 835 MVA steam turbine generator ....................... 67
4.28 Torque-rotor angle characteristics during a step increase in input torque of a 835 MVA steam turbine generator .................. 67
4.29 Single-machine infinite-bus equivalent network .................................................................................. 71
4.30 Thyristor excitation system with AVR and PSS ................................................................................. 71
<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.31</td>
<td>Multilevel MATE partitioning of the single machine infinite bus system</td>
<td>73</td>
</tr>
<tr>
<td>4.32</td>
<td>Transient response of the single-machine infinite-bus system with manual control: (a) terminal voltage, (b) active power, (c) rotor angle</td>
<td>75</td>
</tr>
<tr>
<td>4.33</td>
<td>Transient response of the single-machine infinite-bus system with AVR/PSS: (a) terminal voltage, (b) active power, (c) rotor angle</td>
<td>76</td>
</tr>
<tr>
<td>5.1</td>
<td>Schematic of a doubly fed induction generator wind turbine system</td>
<td>81</td>
</tr>
<tr>
<td>5.2</td>
<td>Two-mass representation of a DFIG WTG shaft for accurate simulation of torsional shaft oscillations</td>
<td>82</td>
</tr>
<tr>
<td>5.3</td>
<td>Rotor and stator-side PWM converters of a DFIG</td>
<td>84</td>
</tr>
<tr>
<td>5.4</td>
<td>Geometric representation of the relationship between stationary abc and αβ reference frame stator voltages, and definition of stator voltage oriented dq reference frame</td>
<td>86</td>
</tr>
<tr>
<td>5.5</td>
<td>Stator-side converter controller</td>
<td>88</td>
</tr>
<tr>
<td>5.6</td>
<td>Geometric representation of the relationship between stationary abc and αβ reference-frame flux linkages, and definition of stator-flux-oriented dq reference frame</td>
<td>90</td>
</tr>
<tr>
<td>5.7</td>
<td>Rotor-side converter controller</td>
<td>93</td>
</tr>
<tr>
<td>5.8</td>
<td>Wind turbine characteristics with maximum energy capture curve</td>
<td>96</td>
</tr>
<tr>
<td>5.9</td>
<td>DFIG wind turbine test case</td>
<td>99</td>
</tr>
<tr>
<td>5.10</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) rotor angular velocity, (b) electromagnetic torque, (c) rotor speed, (d) stator flux</td>
<td>100</td>
</tr>
<tr>
<td>5.11</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) DFIG active power, (b) DFIG reactive power, (c) machine stator active power, (d) machine stator reactive power</td>
<td>101</td>
</tr>
<tr>
<td>5.12</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator-side converter active power, (b) stator-side converter reactive power, (c) rotor-side converter (machine rotor) active power, (d) rotor-side converter (machine rotor) reactive power</td>
<td>102</td>
</tr>
<tr>
<td>5.13</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) DFIG stator-side RMS voltage, (b) DFIG stator-side RMS current, (c) DFIG rotor-side RMS voltage, (d) DFIG rotor-side RMS current</td>
<td>103</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>-----------------------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>5.14</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator phase to neutral voltage, (b) stator phase current, (c) rotor phase to neutral voltage, (d) rotor phase current, (e) stator-side converter phase voltage, (f) stator-side converter phase current</td>
<td>104</td>
</tr>
<tr>
<td>5.15</td>
<td>Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator voltages $dq$ components, (b) stator currents $dq$ components, (c) rotor voltages $dq$ components, (d) rotor currents $dq$ components, (e) stator-side converter voltages $dq$ components, (f) stator-side converter currents $dq$ components</td>
<td>105</td>
</tr>
<tr>
<td>5.16</td>
<td>Comparison of a transient response of the DFIG wind turbine to a step decrease in wind velocity against the experimental results from the literature: (a) rotor speed, (b) d-axis rotor current, (c) q-axis rotor current</td>
<td>106</td>
</tr>
<tr>
<td>5.17</td>
<td>Transient response of the DFIG wind turbine to a three-phase short circuit: (a) rotor angular velocity, (b) electromagnetic torque, (c) rotor speed, (d) stator flux</td>
<td>109</td>
</tr>
<tr>
<td>5.18</td>
<td>Transient response of the DFIG wind turbine to a three-phase short circuit: (a) DFIG active power, (b) DFIG reactive power, (c) machine stator active power, (d) machine stator reactive power</td>
<td>110</td>
</tr>
<tr>
<td>5.19</td>
<td>Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator-side converter active power, (b) stator-side converter reactive power, (c) rotor-side converter (machine rotor) active power, (d) rotor-side converter (machine rotor) reactive power</td>
<td>111</td>
</tr>
<tr>
<td>5.20</td>
<td>Transient response of the DFIG wind turbine to a three-phase short circuit: (a) DFIG stator-side RMS voltage, (b) DFIG stator-side RMS current, (c) DFIG rotor-side RMS voltage, (d) DFIG rotor-side RMS current</td>
<td>112</td>
</tr>
<tr>
<td>5.21</td>
<td>Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator phase to neutral voltage, (b) stator phase current, (c) rotor phase to neutral voltage, (d) rotor phase current, (e) stator-side converter phase voltage, (f) stator-side converter phase current</td>
<td>113</td>
</tr>
<tr>
<td>5.22</td>
<td>Transient response of the DFIG wind turbine DC-link voltage to a three-phase short circuit</td>
<td>114</td>
</tr>
</tbody>
</table>
List of Figures

5.23 Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator voltages' $dq$ components, (b) stator currents' $dq$ components, (c) rotor voltages' $dq$ components, (d) rotor currents' $dq$ components, (e) stator-side converter voltages' $dq$ components, (f) stator-side converter currents' $dq$ components .......................... 115

5.24 Transient response of the DFIG wind turbine to a three-phase short circuit: rotor phase voltages and currents ........................................ 116

5.25 Transient response of the DFIG wind turbine to a three-phase short circuit obtained with the stability tool TSAT: (a) DFIG stator terminal voltage, (b) DFIG rotor angular velocity, (c) DFIG active power, (d) DFIG reactive power .................................................. 117

5.26 Transient response of the DFIG wind turbine to a three-phase short circuit obtained with the stability tool TSAT: (a) DFIG rotor $dq$ voltages and currents (from top to bottom - $v_{dr}$, $i_{dr}$, $v_{qr}$ and $i_{qr}$), (b) DFIG stator current .......................................................... 118

A.1 Illustration of integration with the trapezoidal rule .................. 130

C.1 Circuit configuration for filtering 60 Hz component of $\psi_a(t)$ ........ 134
C.2 Variation of filtered flux-linkage magnitude as a function of frequency 135
C.3 Equivalent circuit of the parallel RLC filter discretized with the trapezoidal rule .......................................................... 136
Acknowledgements

I have been extremely lucky to meet and to get to know so many wonderful people since my arrival in Canada almost eight years ago. I would sincerely like to thank every one of you who has crossed my path on this long journey.

My sincere gratitude goes to my supervisor, Prof. José R. Martí, whose extensive knowledge and brilliant ideas have inspired this work. I thank him for his continuous support and understanding, especially during the difficult times. I would also like to express my sincere gratitude to Prof. Hermann W. Dommel, whose course on electromagnetic transients opened my eyes to the world of power systems simulation. His willingness to help and his invaluable experience is very much appreciated. I would like to acknowledge Prof. Luis R. Linares for introducing me to OVNI and helping me get started with the tremendous task of programming the simulator. His kindness and good heart have helped me endure through many difficult times. I am extremely grateful to Dr. Prabha Kundur, who provided me with many opportunities at Powertech Labs, and helped me see the bigger picture of many problems we have discussed. His guidance and understanding have been invaluable. Many thanks to Dr. Ali Moshref for all the time he spent helping me learn about stability simulations.

A special place in my heart belongs to my dear friends who shared the ups and downs of graduate student life with me. My special thanks goes to my dear friend Richard Rivas, for all the discussions about work and life, as well as his honesty and kindness. I am grateful to my friend Khosro Kabiri for always being there for me. I would like to thank Stephane Bibian, Benedito Bonatto, Jesús Calviño, Maninder Dhaliwal, Jorge Hollman, Awad Ibrahim, Hee-Sang Ko, Daniel Lindemeyer, Fernando Moreira, Dai Nan, Ken Wicks and many others in the old group for their friendship and care. I would also like to thank Meliha Selak for all the help she provided to me and my family along the years. In the new Power Group, I would like to give special thanks to Tom de Rybel for always being so helpful and for organizing fun events. Thanks to Hafiz, Lucy, Quanhong, Marcelo, Michael and everyone else; your thoughts and comments have been appreciated.

I would like to thank my parents, Mira and Ivan, my sister Danijela and my brother Dražen for their love and their incredible support through all of my endeavors. They have inspired me throughout my life to excel in everything I do.

I would like to thank my wonderful husband Gordon and my son David for always
Acknowledgements

being there for me, for their love and patience. You fill my life with joy and happiness.

Finally, I would like to acknowledge the following people and institutions whose invaluable financial support over the years have enabled me to complete this thesis work: Dr. Martí through a research assistantship; UBC through a University Graduate Fellowship and a Li Tze Fong Memorial Fellowship; and Powertech Labs through a research assistantship and several part-time contracts.

Mažana Armstrong, October 2006
Chapter 1

Introduction

1.1 Background

Object Virtual Network Integrator (OVNI) is a simulation tool aimed at obtaining very fast, real-time solutions of large power system networks [1], [2], [3]. OVNI is based on the EMTP program (Electromagnetic Transients Program), used to simulate electromagnetic transients [4]. Quoting from [5], electromagnetic transients "occur on a scale of microseconds (e.g., initial transient recovery voltage), milliseconds (e.g., switching surges), or cycles (e.g., ferroresonance). By nature, these phenomena are a combination of traveling-wave effects on overhead lines and cables, and of oscillations in lumped-element circuits of generators, transformers, and other devices." The term electromagnetic transients is used to distinguish them from the electromechanical oscillations of generators in transient stability studies.

In the EMTP, basic lumped circuit components, such as inductors and capacitors, are described with ordinary differential equations. Differential equations are discretized using the trapezoidal rule of integration (Appendix A) to obtain algebraic difference equations. Discrete time step sizes depend on the type of phenomena studied and the highest frequency expected in the results. They range from a few nanoseconds for very fast transients to an upper limit of about one millisecond dictated by the steady-state base system frequency of 50 or 60 Hz. The EMTP methodology normally requires a three-phase system representation as opposed to the positive sequence single-phase representation used in stability programs which assume that voltages and currents can be represented at the base system frequency as phasors.

The system of equations solved in the EMTP for a network with \( n \) nodes is formed as:

\[
[G] \cdot [v(t)] = [h(t)]
\]  

(1.1)

where

- \([G]\) is a nodal conductance matrix,
- \([v(t)]\) is a vector of nodal voltages,
- \([h(t)]\) is a vector of accumulated currents representing independent current sources and history terms.
The solution for nodal voltages is obtained by first triangularizing the \([G]\) matrix through a Gaussian reduction procedure and then backsubstituting for the updated values of \([h(t)]\). Sparsity techniques\(^1\) are used to reduce the number of operations.

It is apparent even from this very rudimentary description of the EMTP and its applications that a major drawback of EMTP simulations is their speed. The first steps in creating the new OVNI simulator were aimed at “speeding-up” the EMTP simulation [2]. The solution in [2] offered a software approach to topological decoupling of power system networks, using the transmission line models in the EMTP, and a hardware approach to parallelize computation and achieve real-time performance. Soon after this solution, the new concept of MATE emerged, which eliminated the need for decoupling using the transmission line model [7]. Object-oriented software implementation of OVNI was proposed in [8]. Finally, cluster-based hardware solutions for real-time simulations in OVNI were proposed in [9], [10].

### 1.2 Motivation and Objectives

As stated in the previous section, the original idea behind OVNI was to speed up the EMTP simulation, for the following two reasons: (i) to allow larger or more computationally demanding systems to be simulated with the EMTP at acceptable speeds in an off-line environment, and (ii) to allow real-time testing of power system equipment in a closed-loop environment. By combining the concept of system partitioning (Diakoptics\(^2\)) with the concept of system reduction (Thévenin equivalents\(^3\)), the MATE (Multi-Area Thévenin Equivalent) methodology has opened the door to new horizons of investigation of computational procedures and modelling techniques used in simulating large power systems.

Prior to the beginning of this work OVNI existed as a special purpose simulator built for specific tasks to test for real-time performance, as described in [3]. The ultimate goal, however, was and is to develop OVNI as a general purpose power system simulator capable of handling any transient time-domain power system simulation. With that in mind, an object-oriented approach to software programming of OVNI was proposed in [8].

The initiative for the work presented in this thesis started with the notion of building a general purpose simulator. The simulator would, along with the hardware solutions for speed, provide engineers with an all-in-one tool capable of performing

\(^1\)Sparsity applications in power system analysis were first introduced by N. Sato and W. F. Tinney in 1963 [6].

\(^2\)The concept of Diakoptics was first introduced by Gabriel Kron in 1952 [11].

\(^3\)The concept of a Thévenin equivalent was first introduced by Hermann von Helmholtz in 1853 [12]. In 1883, Léon Charles Thévenin, unaware of Helmholtz’s work, published the same result [13], [14].
Chapter 1. Introduction

the simulation of electromagnetic (EMTP-type) transients, as well as electromechanical (stability) transients, with a wide range of available models of power system components. The model complexity would range from detailed models of, for example, frequency-dependent transmission lines and phase-domain electric machine models, to the approximate, simplified classical machine and pi-circuit transmission line representations.

Justification for an integrated approach to power system analyses comes from the trends that have been emerging in the last decades: new technologies involving power electronics equipment (HVDC transmission, FACTS devices) and controls that make standard stability analyses much more complex. An example of a small-scale system that embeds the complexity and challenges of today's power systems is, for example, a wind turbine generation system consisting of an induction generator, two back-to-back pulse-width-modulated (PWM) voltage source inverters, and complex controls to regulate the turbine's responses. Ideally, engineers studying integration of large wind farms into the power system grid would be able to choose the complexity of the models for their simulation, and perform their analyses at different levels of detail for the most accurate picture of an overall state of the system. Ultimately, a combination of detailed and approximate modelling with an overall goal of fast and accurate simulation results would lead to a more reliable design and operation of the system.

The objectives of this work are closely related to the development of the OVNI software, including the computational methods for system partitioning and modelling of power system components. The solutions presented in this thesis, however, are general and can be applied to many existing programs that simulate power system networks.

1.3 Thesis Organization

This thesis is divided into six chapters. The first chapter briefly describes the sequence of events that led to the realization of this work, and states the objectives of this research.

The second chapter provides background information on power system simulation and analysis tools in use today, then concentrates on a literature overview of the methods used for solving large systems of equations associated with the simulation of power system networks. We provide an overview of Sparsity, Diakoptics and Modified Nodal Analysis, and describe their applications.

In Chapter 3, we first review the concept of MATE and introduce the new concept of Multilevel MATE. Multilevel MATE allows multilevel partitioning of the system of equations, enabling the system to be divided according to its nature into changing,
constant and nonlinear subsystems. Multilevel MATE introduces the concept of a functional sublink that allows convenient modelling of power system components that are better described by branch equations. We discuss the computational efficiency of the multilevel approach, as well as the solution of nonlinearities. We provide examples of modelling an ideal switch and a nonlinear control function to demonstrate the concept. Using a block diagram of Multilevel MATE, we depict the computation requirements and the flow of information from the lowest level to the highest level of system partitioning.

In Chapter 4, we describe modelling with Multilevel MATE, and present a general approach to modelling ideal voltage sources connected node-to-ground or node-to-node as sublinks. We provide a solution for matrix computations with Multilevel MATE in the case of ungrounded circuits and ungrounded subsystems, and demonstrate it through examples. We describe and test an extensive and complete derivation of a phase-domain induction machine model implementation with Multilevel MATE, and show the results obtained with a similarly derived phase-domain synchronous machine model. Finally, we describe and show the implementation of a controller’s equations with Multilevel MATE on a standard transients stability test case of a synchronous machine connected to an infinite bus modelled in dq coordinates.

In Chapter 5, we test our modelling methodology and computational solutions on a complex doubly fed induction generator (DFIG) wind turbine system reproduced from an experimental test case described in the literature. Following a brief introduction to wind turbine technology, we provide a detailed description of the test case and the modelling of its components, then perform the tests to validate the modelling. We then compare our results against the experimental results in the literature.

In Chapter 6, the final chapter of the thesis, we state the main contributions of our research and provide recommendations for future work.

1.4 Thesis Contributions

This thesis describes novel power system modelling and implementation solutions suitable for integrated analyses of electromagnetic and stability transients in the newly developed, EMTP based, general purpose simulator OVNI. The main contributions of this thesis can be summarized as follows:

- The new concept of Multilevel MATE is proposed. It improves computational speed of the existing single-level MATE partitioning approach, and allows modelling of power system components with branch equations.

- An efficient framework for iterating nonlinear equations using the Multilevel MATE concept is proposed.
Chapter 1. Introduction

- A new technique for conditioning of ungrounded or weakly grounded circuits and subsystems is presented and demonstrated on examples.

- Branch modelling of power system components using Multilevel MATE is proposed. Induction and synchronous machines, ideal voltage sources, ideal switches and controllers are modelled and then validated through an extensive example of a doubly fed induction generator wind turbine system.

- The new concepts described in this thesis, as well as numerous new power system components models, have been implemented into a new version of the general purpose simulator, OVNI.
Chapter 2

An Overview of Power System Simulation Tools and Techniques

2.1 Introduction

Computer simulation of power systems is a necessary tool for both planning and everyday operation of electric power networks. In the early days of power system analysis, the need for computational aids in analyzing power system networks led to the design of the first analog computer, as early as 1929 [15]. The AC network analyzer was able to determine power flows and voltages during normal and emergency conditions and switching operations. The earliest application of digital computers to power system problems dates back to the 1940s; however, it was rather limited in scope due to the limited capacity of the punch card computers used then. Large-scale digital computers became available in the mid-1950s, and the success of load-flow programs led to the development of programs for short-circuit and stability analyses.

Phenomena analyzed with different simulation methods and tools range from slower widespread network events, such as voltage instability, to fast electromagnetic transients affecting networks locally due to, for example, surges from lightning strikes. Methods for power system simulation range from load-flow [16] programs that use approximate, simpler network representations, to highly sophisticated and complex modelling by the Electromagnetic Transients Program (EMTP) [4].

Power-flow calculations are used to determine the steady-state of the power generation/transmission system for the given loading condition. The power and voltage constraints make the load-flow problem nonlinear. The numerical solution therefore must be iterative, leading to convergency and other solution problems. Possibly thousands of load-flow methods have been developed over the years, streaming from continuous efforts to improve the original load-flow concept and address the needs of, for example, power system reliability and security.

The stability of power systems was associated early on with rotor-angle stability of synchronous machines. More recently, voltage stability has emerged as one of the major concerns in modern power systems characterized by highly loaded transmission networks operating close to their capacity limits. Dynamic simulations aimed at
studying the stability of power system networks require more complex modelling than do power-flow studies. Dynamic simulations are currently an integral part of planning and operation studies of power system networks.

Electromagnetic transients simulations are the most common methods for simulation of power system surges and fast-circuit transients. Electromagnetic transients are rarely an important factor in system planning or operating decisions, but are essential factors in the design process. These types of simulations require extensive modelling detail and small simulation time steps due to the small time constants associated with the studied phenomena. With the ever-increasing computer power available today, the EMTP has become accepted for the solution of a wide range of transient problems. In particular, the EMTP-type programs are used extensively to analyze system disturbances in power system protection design.

Regardless of the simulation method used and the complexity of the associated modelling, most of the tools employed today describe power system transmission networks with nodal equations that are particularly suitable for digital simulation. Moreover, they all use similar programming techniques to efficiently handle large matrices. The following section discusses the historical development of these techniques and their applications.

### 2.2 Techniques for Simulation of Large Power System Networks

Nodal analysis implements Kirchhoff's current law, which states that the sum of the currents entering the node in a circuit is equal to the sum of the currents leaving the same node. As its name suggests, nodal analysis produces nodal (node) voltages as a solution. The nodal approach has an advantage over the mesh approach derived from Kirchhoff's voltage law in that it introduces fewer variables and equations in the case of power system networks, therefore leading to smaller system matrices. The system of equations associated with nodal analysis has the general form:

\[
[Y] \cdot [V] = [I] \tag{2.1}
\]

where

- \([Y]\) is a nodal admittance matrix,
- \([V]\) is a vector of nodal voltages,
- \([I]\) is a vector of nodal currents.

---

*Kirchhoff's laws were first announced by Gustav Robert Kirchhoff in 1845.*
Chapter 2. An Overview of Power System Simulation Tools and Techniques

To solve this system of equations for nodal voltages, one has to determine the inverse of \([Y]\). For a large number of nodes, this task is extremely demanding computationally, especially in the past when computer power was limited and expensive compared to today. Practically from the beginning of the digital-computer simulation era, power systems engineers have been developing techniques to enable faster solutions of large systems of equations. Two main very distinctive streams established themselves in the 1960s, Diakoptics and Sparsity Techniques. Sparsity techniques, highly favoured over Diakoptics, have developed roots in the majority of the simulation tools in existence today.

To overcome the limitations of nodal analysis in the modelling of some system components, modified nodal analysis (MNA) was formulated in the mid-1970s. Because of its relevance to this work, we will discuss MNA in the last part of this section.

2.2.1 Sparsity

In 1978, W. F. Tinney, referring to his previous papers [6] and [17] wrote in a commentary: “The aim of sparsity methods is to exploit the property that the coefficients of many large systems of simultaneous equations are mostly zero. Standard matrix methods take no advantage of the zeros, storing and processing them the same as non-zero numbers, and causing most, or all, of them to become non-zero in the solution process... Today, awareness of sparse matrix methods is practically inescapable. Papers, books, and symposia on sparsity abound and applications exist in almost every field. But only a decade ago, when our paper was published, the concepts of sparsity were virtually unknown. The advantages are so great (10 to 100 or more) and the idea is so simple, it is puzzling that it took so long to be discovered.” [18]

The first paper on sparsity appeared in 1957 in Management Science [19]. The first sparsity method applied to the solution of power system networks was developed by Sato and Tinney in 1961, and published in 1963 [6]. The work in these papers was not recognized or accepted after published, and remained unknown until the mid 1960s. In 1967, when the second paper was published by Tinney and Walker [17], sparsity was finally acknowledged, and has since become widely accepted and applied in power system simulation tools. The original method proposed in the 1960s remains valid today, with most improvements consisting of marginally more efficient algorithms. By 1977 when a survey paper on sparse matrix research was published [20], there were already over six hundred published references on the topic of sparsity.

In nodal analysis the power system network is described by a large sparse matrix of coefficients referred to as its “admittance matrix”. The task of solving the system for nodal voltages involves obtaining the inverse of the matrix of coefficients, which is a very computationally demanding task for large power systems including 20,000
to 50,000 nodes. By appropriately ordered triangular decomposition, the inverse of a sparse matrix can be expressed as a product of sparse matrix factors, which brings an advantage in computational speed, storage requirements and accuracy. The method of sparsity follows these essential steps: (1) store and process only non-zero terms; (2) factor out the coefficient matrix into upper and lower triangular matrices; (3) order the factorization to approximately minimize new non-zero terms; and (4) use the factors to obtain the direct solutions by forward and back substitution.

The most evident advantage that can be taken of in sparse systems is information storage and retrieval. This aspect alone allowed simulation of large systems with more accurate models. The computation procedures are organized to avoid operations with zeros (or ones), to minimize overhead such as accessing of arrays and to reduce the arithmetic operations involved. Sparsity algorithms are designed to preserve the system's initial sparsity as much as possible.

The main drawbacks of sparsity methods are related to implementation issues. In order to achieve the advantages of sparsity, the programming scheme must store and process only non-zero elements. In a matrix, the address of each element in the computer's memory can be related to its row and column indices. By using sparsity, in addition to the memory allocation of non-zero matrix elements, tables of indexing information are required to identify the elements and facilitate their addressing. This procedure, if not properly planned, may diminish the time-saving advantages of sparsity methods.

The most effective way of ordering the operations is performed as if it were being done by visual inspection and manual computation. It is very important to realize that, although the word "optimal" is often used to describe ordering algorithms, no known algorithms are optimal in a global sense for general sparse matrices. At the completion of the ordering algorithm, the exact form of the table of factors is established, and this information is recorded to guide the elimination process. During the elimination, no operation is performed that would lead to a predictable zero result, and no memory is allocated for a predictable zero element. The original matrix, the table of factors, and all indexing tables contain only non-zero elements.

Rows from the original matrix are transferred to a compact working row in which elements to the left of the diagonal are eliminated by appropriate linear combination with previously processed rows from the partially completed table of factors. When work on the row has been completed, it is added to the table of factors. For nonsymmetric matrices, the derived elements to the left of the diagonal must be stored as well. Finally, a direct solution of the system of equations is obtained by multiplying the matrices derived from the table of factors.
2.2.2 Diakoptics

In Diakoptics, the networks are partitioned into subnetworks that can be analyzed independently, then the subnetwork solutions are combined to obtain the solution of the entire network. In the literature, Diakoptics is also referred to as a solution of networks by tearing. The concept of Diakoptics was first proposed by Gabriel Kron in 1953 [11], in the *Journal of Applied Physics*. His sixteen-page paper describes the advantages of system tearing and gives hints about a multi-machine solution of large-system matrix equations. It also points out a reduction in the computation time of matrix inversion on a single machine simply due to partitioning. His brilliant ideas, which are in fact one of the essences of the OVNI simulator, were well ahead of his time and were not understood by his contemporaries.

Kron’s Diakoptics, nevertheless, was not completely forgotten. A number of publications expanded on his ideas for applications in analysis of power systems, electrical networks and structural and other large-scale systems. In Kron’s original form, Diakoptics required the explicit computation of inverses of submatrices representing the individual subnetworks. As a result, sparsity of subnetwork equations could not be exploited. A number of works published in the 1970s, just after sparsity established itself, investigated network analysis by tearing using sparse matrix solution techniques. A good overview of these attempts is given in [21].

To this day, however, Diakoptics, however, has been questioned as to whether more efficient computations are possible with the use of sparsity and solving the network as a whole. The only situations when system tearing was deemed necessary involved very large networks, the equations for which could not be stored on a single computer, even though sparse matrix solution techniques were being used. Tearing was also recognized as necessary in cases of repetitive identical networks when the equations of only one such network needed to be solved and stored. In addition, Diakoptics was identified by only a few researchers as a solution to allow parallel processing and the use of latency concepts in finding the network solution.

Possibly the main reason why Diakoptics never developed roots in the simulation of large power system networks was due to its ambiguous description in the many papers that could not deliver clear application examples and justify its use. Moreover, a number of papers, as stated in [20], claimed that Diakoptics tearing could never be more advantageous than sparsity techniques.

2.2.3 Modified Nodal Analysis

As stated in [22], nodal analysis “treats voltage sources inefficiently and is incapable of including current-dependent elements, linear or nonlinear... Another disadvantage of the nodal method is that branch currents are not accurately or conveniently obtained
as part of the output of the program.” In this 1975 paper, the new concept of modified nodal analysis (MNA) is proposed, which allows branch equations to be incorporated into the system of nodal equations, eliminating these disadvantages.

Certain circuit components, such as voltage sources, transformers and dependent sources, cannot be incorporated into nodal analysis in a natural way. In addition, certain variables, such as source or transformer currents, have to be obtained by postprocessing the results of nodal analysis. In the modified nodal analysis, all non-natural elements are removed in order to write a conventional set of nodal equations, and are then reintroduced with branch equations.

![Independent voltage source](image)

Figure 2.1: Independent voltage source

The system of equations associated with MNA can be demonstrated through an example of an ideal voltage ($V_s$) and series impedance ($Z_s$) connected between the nodes $a$ and $b$ in Fig. 2.1. The original set of nodal equations (2.1) is affected in two ways. First, the new constraint equation is introduced relating the two nodal voltages, $V_a$ and $V_b$. Second, the new unknown current, $I_s$, is permitted to flow. The new equation will increase the number of rows and columns in the system matrix by one. Taking into account the flow of the current and the polarity of the voltage source, as depicted in Fig. 2.1, the following MNA system of equations is obtained:

$$
\begin{bmatrix}
Y_{R} & +1 \\
+1 & -1 & -Z_s
\end{bmatrix}
\begin{bmatrix}
V_a \\
I_s
\end{bmatrix}
= 
\begin{bmatrix}
I_a \\
V_s
\end{bmatrix}
$$

(2.2)

where $I_a$ and $I_b$ are the nodal injection currents of the system.

In (2.2), matrix $[Y_R]$ is a reduced form of the nodal matrix $[Y]$, excluding the contribution due to voltage sources, current controlling elements, etc. For any given circuit, the MNA matrix dimension is the sum of the number of nodes and the number of currents as outputs.

In situations where the series impedance $Z_s$ is zero, conventional nodal analysis would not produce a solution because of an undefined $1/Z_s$ entry into the admittance matrix, unless one of the nodes, $a$ or $b$, is reduced from the system of equations. With the modified nodal analysis, a solution can be found by employing a suitable pivoting strategy to maintain the strong diagonal of the MNA matrix, which is desirable from
an accuracy and efficiency point of view. The MNA equations are ordered in such a way as to reduce both execution time and storage requirements.

The row interchange process is applied to the node equation of each node connected to one or more voltage sources. The node equation is always interchanged with one of the branch equations having the smallest number of non-zeros to minimize fill-ins. For example, in (2.2) the bottom equation would be interchanged either with the node equation for \( V_a \) or \( V_b \), depending on their number of non-zero elements dictated by the composition of the adjacent network.

Modified nodal analysis combined with sparsity techniques is commonly used in today's simulation tools. It removes the limitations imposed by classical nodal analysis and is appropriate for the symbolic and numeric analysis of linear electrical circuits. Adopting MNA results in an increased order of the system of equations. These branch equations are generally highly sparse and can be efficiently dealt with using modern sparsity-based computing packages.

### 2.3 Summary

This chapter provides a brief overview of the existing methods and tools for power system simulation. Further discussion on existing techniques for efficient computation of large and sparse systems of equations, namely Sparsity and Diakoptics, is presented. Modified nodal analysis and its advantages is also presented.
Chapter 3

Multilevel MATE

3.1 Introduction

In this chapter, we introduce our novel concept of Multilevel MATE. Multilevel MATE expands the original concept of MATE by allowing subsystems to be partitioned with sublinks. A subsystem, for example, can be partitioned into constant and changing nature sub-subsystems, where only the changing nature subsystem’s matrix needs to be recalculated at every simulation time step. An example of a power system element of changing nature is a phase-domain synchronous machine model.

Multilevel MATE also allows for convenient modelling of power system components whose function is better described by branch equations, as opposed to the nodal equations typically used in the EMTP. An example of such an element is a switch. If the change of a system’s topology due to switching is confined to branch equations, the subsystem’s nodal equations are unaffected. Subsystem partitioning and branch (functional) modelling bring a significant improvement to the simulation efficiency of the OVNI code.

Multilevel MATE provides an efficient framework for implementing nonlinearities that require an iterative procedure. Power system components that can be conveniently modelled within the Multilevel MATE framework include switches, controllers, transformers, synchronous and induction machines, voltage-source inverters etc.

The research contributions reported in this chapter include the following:

- *Formulation of the new Multilevel MATE concept.*
- *Functional modelling of power system components with Multilevel MATE.*
- *Description of the new and efficient framework for iterating nonlinear equations with Multilevel MATE.*
3.2 MATE System Partitioning

The Multi-Area Thévenin Equivalent (MATE) concept was first introduced by Martí [23] in an internal report to his research group. Reference [1] presents a detailed explanation of the algorithm MATE that extends the main ideas of Diakoptics by recognizing that the subsystems split by the branch links can be represented by Thévenin Equivalents. To demonstrate the MATE concept, let us consider the system in Fig. 3.1. Any system can be partitioned into subsystems by introducing links. For example, the system depicted in Fig. 3.1 is partitioned into three subsystems A, B and C, by introducing six links. The hybrid system of modified nodal equations has the following form:

\[
\begin{bmatrix}
A & 0 & 0 & p \\
0 & B & 0 & q \\
0 & 0 & C & r \\
[p^t \ q^t \ r^t \ -z]
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
i_\alpha
\end{bmatrix}
= 
\begin{bmatrix}
h_A \\
h_B \\
h_C \\
-V_\alpha
\end{bmatrix}
\]

where:

- \([A], [B] \text{ and } [C]\) are the subsystems' admittance matrices
- \([p], [q] \text{ and } [r]\) are the subsystems' current injection (or connectivity) arrays
- \([p^t], [q^t] \text{ and } [r^t]\) are transpose arrays of \(p, q \text{ and } r\)
- \([z]\) is a matrix of the links' Thévenin impedances
Chapter 3. Multilevel MATE

- \([h_A], [h_B], [h_C]\) are vectors of the subsystems' accumulated currents
- \([V_a]\) is a vector of the links' Thévenin voltages
- \([v_A], [v_B], [v_C]\) are the subsystems' nodal voltages
- \([i_a]\) is a vector of the links currents.

Connectivity arrays for the six links connecting the subsystems are constructed so that a particular link current is assigned a direction of flow where the "from" node is assigned a "1" in the connectivity array of the "from" subsystem, and the "to" node is assigned a "-1" in the connectivity array of the "to" subsystem. Any component (or branch) in the system can become a link described by its Thévenin impedance \([z]\) and its Thévenin voltage \([V_a]\). The most general representation of a link is depicted in Fig. 3.2. Note that \([z]\) and/or \([V_a]\) can be zero.

![Figure 3.2: General description of a link](image)

Partitioned matrices in (3.1) are manipulated to obtain the subsystems' Thévenin equivalents as seen from the linking nodes [1]. The result of the manipulation is the MATE system of equations, in the following form:

\[
\begin{bmatrix}
I & 0 & 0 & a \\
0 & I & 0 & b \\
0 & 0 & I & c \\
0 & 0 & 0 & z_a
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
i_a
\end{bmatrix}
= 
\begin{bmatrix}
e_A \\
e_B \\
e_C \\
e_a
\end{bmatrix}
\]

where

\[
a = A^{-1}p \\
b = B^{-1}q \\
c = C^{-1}r \\
z_a = p^t a + q^t b + r^t c + z \\
e_a = p^t e_A + q^t e_B + r^t e_C + V_a
\]

and \([I]\) is the identity matrix.

The individuality of each subsystem is preserved in (3.2). Vector \([e_A]\) and array \([a]\) represent, respectively, the Thévenin equivalent voltage vector and the Thévenin
impedance array of subsystem $A$ as seen from the linking nodes. The solution for link currents is now independent from the solution for the nodal subsystems’ voltages. The interaction of the subsystems’ Thévenin equivalents is solved at the links level and returned to each subsystem in the form of injected link currents at the linking nodes:

$$i_{\alpha} = z_{\alpha}^{-1} e_{\alpha}$$

The subsystems’ nodal equations are then solved independently from each other at the subsystem level as:

$$v_A = e_A - a \cdot i_{\alpha}$$
$$v_B = e_B - b \cdot i_{\alpha}$$
$$v_C = e_C - c \cdot i_{\alpha}$$

### 3.3 Multilevel MATE Concept

The concept of Multilevel MATE introduced in this thesis can be demonstrated on the same system depicted in Fig. 3.1. In this case, the system is further divided into ten sub-subsystems with a total of six links and twenty sublinks, as shown in Fig. 3.3. The complete system of equations for the depicted system is shown in (3.3). Matrices $[A]$, $[B]$ and $[C]$ represent the conductance matrices of the corresponding subsystems, while $[p_A], [q_B], [r_C]$ are the connectivity arrays of the sub-subsystems ($A_1, A_2, \cdots B_1, B_2, \cdots C_1, C_2$) within the corresponding subsystems.

Figure 3.3: An example of a system that demonstrates the Multilevel MATE concept
Chapter 3. Multilevel MATE

The hybrid system of equations describing the system shown in Fig. 3.3 has the following form:

\[
\begin{bmatrix}
A & p_A \\
p_A^T & -z_A
\end{bmatrix}
\begin{bmatrix}
p \\
p_A^T
\end{bmatrix}
+ \begin{bmatrix}
p \\
q
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
\nu_{\text{sublink}}
\end{bmatrix}
= \begin{bmatrix}
h_A \\
-V_{A,\text{sublink}} \\
V_B \\
-V_{B,\text{sublink}} \\
V_C \\
-V_{C,\text{sublink}}
- V_\alpha
\end{bmatrix}
\]

where

\[
[A] = \begin{bmatrix}
A_1 & 0 & 0 \\
0 & A_2 & 0 \\
0 & 0 & A_3
\end{bmatrix},
[B] = \begin{bmatrix}
B_1 & 0 & 0 & 0 \\
0 & B_2 & 0 & 0 \\
0 & 0 & B_3 & 0 \\
0 & 0 & 0 & B_4 \\
0 & 0 & 0 & B_5
\end{bmatrix}
\]

\[
[C] = \begin{bmatrix}
C_1 & 0 \\
0 & C_2
\end{bmatrix}
\]

and

\[
p_A = \begin{bmatrix}
p_{A1} \\
p_{A2} \\
p_{A3}
\end{bmatrix},
p = \begin{bmatrix}
p_1 \\
p_2 \\
p_3
\end{bmatrix},
v_A = \begin{bmatrix}
v_{A1} \\
v_{A2} \\
v_{A3}
\end{bmatrix},
h_A = \begin{bmatrix}
h_{A1} \\
h_{A2} \\
h_{A3}
\end{bmatrix}
\]

for subsystem A, with similar expressions for subsystems B and C. Matrices \([z_A]\), \([z_B]\) and \([z_C]\) correspond to the sublink's Thévenin impedances, while vectors \([V_{A,\text{sublink}}]\), \([V_{B,\text{sublink}}]\) and \([V_{C,\text{sublink}}]\) correspond to the sublink's Thévenin voltages. The dimension of vectors \([i_{A,\text{sublink}}]\) and \([V_{A,\text{sublink}}]\) is that of the number of sublinks in subsystem A.

To solve the Multilevel MATE equations, first the MATE concept is applied to
Chapter 3. Multilevel MATE

each subsystem to obtain the system of equations (3.4):

\[
\begin{bmatrix}
I & A^{-1}p_A & 0 & 0 & A^{-1}p \\
0 & p_A A^{-1}p_A + z_A & I & B^{-1}q_B & p_A A^{-1}p \\
0 & 0 & 0 & q_B B^{-1}q_B + z_B & q_B B^{-1}q \\
0 & 0 & 0 & 0 & C^{-1}r \\
[p^t \ 0] & [q^t \ 0] & [r^t \ 0] & [r^t C^{-1}r \ -z]
\end{bmatrix}
\begin{bmatrix}
A^{-1}h_A \\
p_A A^{-1}h_A + V_{A,sublink} \\
b^{-1}h_B \\
q_B B^{-1}h_B + V_{B,sublink} \\
C^{-1}h_C \\
r_C C^{-1}h_C + V_{C,sublink} \\
-V_a
\end{bmatrix}
\]

(3.4)

Note, that the sublink currents for subsystems A, B and C, respectively, are:

\[
(p_A A^{-1}p_A + z_A) \cdot i_{A,sublink} + p_A A^{-1}p \cdot i_a = p_A A^{-1}h_A + V_{A,sublink}
\]

\[
(q_B B^{-1}q_B + z_B) \cdot i_{B,sublink} + q_B B^{-1}q \cdot i_a = q_B B^{-1}h_B + V_{B,sublink}
\]

\[
r_C C^{-1}r_C + z_C \cdot i_{C,sublink} + r_C C^{-1}r \cdot i_a = r_C C^{-1}h_C + V_{C,sublink}
\]

(3.5)

Before applying the MATE concept to the entire system, we note that the sublink currents at the subsystem level in (3.3) do not need to be known at the system level. The computational efficiency of the MATE solution is best maintained if the sublink branch currents are “somehow” made invisible to the top level of the system’s partitioning. Going back to Equation (3.3), we can remove the equations for the sublink currents from the system of equations (3.4) and transfer these currents to the right-hand side, as shown in (3.6).

\[
\begin{bmatrix}
A & 0 & 0 & p \\
0 & B & 0 & q \\
0 & 0 & C & r \\
[p^t \ q^t \ r^t \ -z]
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
i_a
\end{bmatrix}
= \begin{bmatrix}
h_A - p_A \cdot i_{A,sublink} \\
h_B - q_B \cdot i_{B,sublink} \\
h_C - r_C \cdot i_{C,sublink} \\
-V_a
\end{bmatrix}
\]

(3.6)

Equations (3.5) and (3.6) fully describe the system. Equation (3.6) closely resembles the system of original equations for the single-level MATE system partitioning (3.1). Next, the MATE equations (3.2) are applied to (3.6) to obtain the following
Chapter 3. Multilevel MATE

equations:
\[
\begin{bmatrix}
1 & 0 & 0 & a \\
0 & 1 & 0 & b \\
0 & 0 & 1 & c \\
0 & 0 & 0 & z_a
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
i_\alpha
\end{bmatrix}
= 
\begin{bmatrix}
e_A - e_{A,\text{sublink}} \\
e_B - e_{B,\text{sublink}} \\
e_C - e_{C,\text{sublink}} \\
e_\alpha - e_{\alpha,\text{sublink}}
\end{bmatrix}
\]  
(3.7)

where
\[
e_{A,\text{sublink}} = A^{-1}p_A \cdot i_{A,\text{sublink}}
\]
\[
e_{B,\text{sublink}} = B^{-1}q_B \cdot i_{B,\text{sublink}}
\]
\[
e_{C,\text{sublink}} = C^{-1}r_C \cdot i_{C,\text{sublink}}
\]
\[
e_{\alpha,\text{sublink}} = p^t e_{A,\text{sublink}} + q^t e_{B,\text{sublink}} + r^t e_{C,\text{sublink}}
\]

Vectors \([e_{A,\text{sublink}}]\), \([e_{B,\text{sublink}}]\) and \([e_{C,\text{sublink}}]\) represent the contributions of the sublink currents to the Thévenin equivalent voltages of each subsystem as seen from the linking nodes. By substituting (3.5) into (3.8), we can eliminate the sublink currents from the overall system of equations and obtain modified Thévenin impedances and voltages. For subsystem \(A\), for example, \([a]\) and \([e_A]\) represent a Thévenin equivalent as seen from the linking nodes, as if the sublinks were not present in the system. The Modified Thévenin Equivalent that includes the sublinks’ contributions is represented by \([a_{\text{MTE}}]\) and \([e_{A,\text{MTE}}]\).

The system of equations for the Modified Thévenin Equivalents is:
\[
\begin{bmatrix}
1 & 0 & 0 & a_{\text{MTE}} \\
0 & 1 & 0 & b_{\text{MTE}} \\
0 & 0 & 1 & c_{\text{MTE}} \\
0 & 0 & 0 & z_{a,\text{MTE}}
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C \\
i_\alpha
\end{bmatrix}
= 
\begin{bmatrix}
e_{A,\text{MTE}} \\
e_{B,\text{MTE}} \\
e_{C,\text{MTE}} \\
e_{\alpha,\text{MTE}}
\end{bmatrix}
\]  
(3.9)

where
\[
a_{\text{MTE}} = a - \Delta a
\]
\[
e_{A,\text{MTE}} = e_A - \Delta e_A
\]
\[
b_{\text{MTE}} = b - \Delta b
\]
\[
e_{B,\text{MTE}} = e_B - \Delta e_B
\]
\[
c_{\text{MTE}} = c - \Delta c
\]
\[
e_{C,\text{MTE}} = e_C - \Delta e_C
\]
\[
z_{a,\text{MTE}} = p^t a_{\text{MTE}} + q^t b_{\text{MTE}} + r^t c_{\text{MTE}} + z
\]
\[
e_{\alpha,\text{MTE}} = p^t e_{A,\text{MTE}} + q^t e_{B,\text{MTE}} + r^t e_{C,\text{MTE}} + V_a
\]

and
\[
\Delta a = A^{-1}p_A (p_A^t A^{-1} p_A + z_A)^{-1} p_A^t \cdot a
\]
\[
\Delta e_A = A^{-1}p_A (p_A^t A^{-1} p_A + z_A)^{-1} (p_A^t \cdot e_A + V_{A,\text{sublink}})
\]

for subsystem \(A\), with similar expressions for subsystems \(B\) and \(C\).

At every time step, each subsystem will pass its modified Thévenin impedances and voltages (i.e., \([a_{\text{MTE}}]\) and \([e_{A,\text{MTE}}]\) for subsystem \(A\)) to the links. The system of links is solved first, and the link currents are returned to their corresponding subsystems to obtain the solution for the subsystem voltages. In the case where subsystem
links are introduced for the purpose of subsystem partitioning (partitioning sublinks), the sublink currents do not need to be solved. However, if some of the sublinks are functional sublinks (e.g., nonlinear elements) and it is important to monitor their currents (as in the case of switches), these sublinks need to be solved at every time step. In the following example, an ideal switch implementation with the Multilevel MATE approach is presented.

### 3.3.1 Example: An Ideal Switch Implementation with Multilevel MATE

Multilevel MATE allows switches to be modelled as sublinks within each subsystem. Switching operations involve a change of the subsystem’s topology. However, this change can be confined to terms in (3.11) that, in effect, modify the subsystem’s Thévenin equivalent as seen from the connecting link nodes. Fig. 3.4 depicts a simple example system to demonstrate the principles of ideal switching on a subsystem level.

![Diagram](image)

Figure 3.4: An example system to demonstrate ideal switching using the Multilevel MATE concept

The system is composed of two subsystems, A and B, connected through a single link. An ideal switch is placed within subsystem A and its branch equation is treated as a sublink. The hybrid system of equations is written from (3.3) in general terms as:

\[
\begin{bmatrix}
A_1 & 0 & p_A^1 \\
0 & A_2 & p_A^2 \\
p_A^t & 0 & -z_A
\end{bmatrix}
\begin{bmatrix}
p_1 \\
p_2 \\
0
\end{bmatrix}
= \begin{bmatrix}
v_A^1 \\
v_A^2 \\
0
\end{bmatrix}
\begin{bmatrix}
i_{A,\text{sublink}} \\
v_B \\
i_\alpha
\end{bmatrix}
= \begin{bmatrix}
h_A^1 \\
h_A^2 \\
-V_{A,\text{sublink}}
\end{bmatrix}
\]

(3.12)
Chapter 3. Multilevel MATE

For the example system it is written as:

\[
\begin{bmatrix}
3 & -2 & 0 & 0 & 0 \\
-2 & 2 & 0 & 0 & 1 \\
0 & 0 & 2 & -2 & -1 \\
0 & 0 & -2 & 3 & 0 \\
0 & 1 & -1 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4 \\
v_5 \\
v_6 \\
\end{bmatrix}
\]

With the switch taken out of the system, we apply MATE as in (3.2), and obtain the subsystems’ Thévenin equivalents as seen from the link nodes:

\[z_a = p^t a + q^t b + z = 1 + 1.25 + 1 = 3.25 \Omega\]
\[e_a = p^t e_A + q^t e_B + V_o = 0 - 1 + 0 = -1 [V]\]

(3.13)

where

\[a = \begin{bmatrix}
A_1^{-1} p_1 \\
A_2^{-1} p_2 \\
\end{bmatrix}, \quad b = B^{-1} q\]
\[e_A = \begin{bmatrix}
A_1^{-1} h_{A1} \\
A_2^{-1} h_{A2} \\
\end{bmatrix}, \quad e_B = B^{-1} h_B\]

The modified Thévenin equivalent is obtained for the case where the switch is closed from (3.10) and (3.11). By referring to (3.12), the case where a switch in subsystem A is closed implies that the corresponding diagonal element of the \([z_A]\) matrix becomes zero \((R_{\text{switch}} = 0)\). For an ideal switch opening, the corresponding column of the subsystem’s connectivity array \([p_A]\) and the row of \([p_A^t]\) are set to zero, and a diagonal of \([z_A]\) is set to one \((i_{\text{switch}} = 0)\). The change of topology due to switching will modify the \(\Delta\) terms in (3.11) or, in a more general case with nonlinearities, the \(\Delta\) terms in (3.17). For example, when the switch is closed we obtain:

\[z_{a,MTE} = (p^t a - p^t \Delta a) + q^t b + z = (1 - 0.33) + 1.25 + 1 = 2.92 \Omega\]
\[e_{a,MTE} = (p^t e_A - p^t \Delta e_A) + q^t e_B + V_o = (0 + 0.33) - 1 + 0 = -0.67 [V]\]

When the switch is open, the modified Thévenin equivalent is the same as (3.13). The switch current can be calculated at any time during the simulation by solving the equation for sublink currents.

Modelling a switch using the Multilevel MATE concept has several advantages over the previous approach used in OVNI [8]. The subsystem admittance matrix (subsystem topology) is not affected by switching, therefore it does not need to be
retriangularized and/or prestored. The size of the subsystem matrix is not increased by adding branch equations that are solved simultaneously with nodal equations, as is the case with the modified nodal analysis. MNA not only increases the size of the subsystem matrix, but also mixes nodal and branch equations. Finally, with the new approach the switch current can be calculated at every time step, or only when needed.

The case of an ideal switch implementation with Multilevel MATE can be extended to any element that contributes to a subsystem with branch equations. Such elements include, for example, ideal transformers, controllers, average models of voltage source inverters, and so on. These branch equations are treated as sublinks in OVNI and contribute to the subsystems' Thévenin equivalents obtained from nodal equations to form Modified Thévenin equivalents (MTEs).

### 3.4 Computational Efficiency of Multilevel MATE Partitioning

The MATE system partitioning is the core feature of OVNI that maps the partitioned network onto a hardware configuration of PC clusters to allow for very fast simulation speeds [9]. Each subsystem and the system of link (branch) equations are solved on separate PC clusters that communicate with each other at a predefined rate. High MATE partitioning of the network produces smaller-sized subsystems that are solved more efficiently (with a higher hardware price), but which also results in an increased number of links that have the opposite effect on the simulation speed. The size of the system of links, because of its communication requirements, is a limiting factor in achieving real-time simulation speeds with OVNI [10]. The aim of MATE partitioning is to attain an optimal compromise between maximizing the computation speed of the subsystems and minimizing the communication with the links system by keeping its size reasonably small. Optimizing system partitioning is beyond the scope of this thesis. However, papers dealing with the subject have been published since the early days of Diakoptics, with the most recent one published in conjunction with the development of OVNI [24].

Multilevel MATE partitioning recognizes the fact that the partitioning (and its corresponding links) that may be required to separate parts of a system due to their different natures (i.e., different simulation step requirements, constant vs. changing component natures) may not in general correspond to optimal partitioning for the most efficient computational time. Moreover, the links' system size, being a constraint for real-time simulation, should not be affected by the presence of, for example, nonlinearities, controllers and switches in subsystems. The Multilevel MATE concept offers an efficient solution for dealing with branch equations at the subsystem level.
without affecting the size of the links or the size of the subsystems’ nodal equations.

Multilevel MATE introduces a new type of link at the subsystem level, referred to as sublinks. Sublinks are divided into two categories, partitioning sublinks, which are introduced to topologically partition a subsystem, and functional sublinks, which come from the modelling requirements of certain power system components. Partitioning sublinks are introduced to increase the efficiency of the computation, for example, by dividing a subsystem into sub-subsystems of constant and changing natures. A good example of a power system component with a changing nature that needs to be updated at every simulation time step is a phase-domain induction or synchronous machine model [25]. Multilevel partitioning also allows for the use of latency [26]. For example, a power electronics device model that requires high frequency switching and therefore a very small time step, μs, can be interfaced through sublinks to the rest of the subsystem, which can be accurately simulated at a slower rate (ms). Functional sublinks are branch equations introduced by the modelling requirements of power system components. For example, a controller or a switch will introduce branch equations. Other power system components that can introduce branch equations include synchronous and induction machines and transformers. Examples are given in subsequent chapters.

The benefits of second-level MATE will be briefly demonstrated on the example shown in Fig. 3.1 and Fig. 3.3. Figure 3.1 depicts a system separated into three subsystems and six links based, for example, on some optimal partitioning algorithm for real-time simulation. It is assumed that each subsystem (A, B, C) is solved on a separate PC cluster processor, and that the links’ system of six links communicates with the subsystems at a predefined rate. In Fig. 3.3, each subsystem is partitioned into sub-subsystems. Because the links’ system is unchanged, the difference in simulation speed between the single level and multilevel approach will occur inside the subsystems.

We have made several assumptions for the purposes of comparing the computational efficiencies of the solution of non-partitioned and partitioned subsystems. The first assumption is that subsystems A, B and C have no functional sublinks. The sublinks that are introduced are only for partitioning purposes. The second assumption is that all of the subsystems are of a changing nature and that their admittance matrices need to be updated at every time step. This situation is the most computationally demanding scenario. We compare the three subsystems assuming the same number of nodes (90, 900, 1800 or 3600), and a different number of partitions and links. Based on Fig. 3.3, Table 3.1 is derived. The result of our comparison of the computational efficiency of the single-level and multilevel approaches in floating point operations (flops) are shown in Fig. 3.5.

Higher partitioning of subsystems leads to faster computational time (comparison between subsystems A, B and C in Fig. 3.5 along y-axis), as long as the sublinks are
Chapter 3. Multilevel MATE

<table>
<thead>
<tr>
<th>Subsystem</th>
<th>Number of Sub-subsystems</th>
<th>Number of Sublinks</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>3</td>
<td>6</td>
</tr>
<tr>
<td>B</td>
<td>5</td>
<td>13</td>
</tr>
<tr>
<td>C</td>
<td>2</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 3.1: Partitioning of subsystems

Figure 3.5: Multilevel MATE vs. single-level MATE computation speedup

not too numerous relative to the size of the sub-subsystems. For example, in the case where subsystems A, B and C have ninety nodes each, the optimal partitioning among the three subsystems is the partitioning of subsystem A with three sub-subsystems of thirty nodes and six sublinks. The speedup over the single-level MATE with no subsystem partitioning is 2.88 times, as opposed to the speedup for partitioning of subsystems B and C of 2.35 and 2.79 times, respectively.

Improvement in the computational speed over single-level MATE is greater for large networks with many nodes (comparison between different number of nodes for individual subsystems in Fig. 3.5 along x-axis). Multilevel MATE reduces the number of computations over twenty times in the case of the largest number of nodes for partitioning of subsystem B. If some of the subsystem matrices were constant by nature, their inverses would be pre-calculated prior to the simulation run time, and the Multilevel MATE partitioning would show an even greater increase in calculation speed. If sub-subsystems $B_1$, $B_2$, $B_3$ and $B_4$ were constant and $B_5$ was the only subsystem of changing nature, Multilevel vs. single-level MATE would result in a computational speedup of over a hundred times for the case of 3600 nodes.
In general, for subsystems comprised of both changing and constant nature elements and for subsystems that are sparse, the application of the Multilevel MATE concept will always lead to a significant computational speedup over the single-level MATE concept. If conventional sparsity techniques were applied for subsystems with changing topology, each topological change (e.g. switching) within a particular region of the subsystem would require the retriangularization of the entire subsystem matrix. With the Multilevel MATE concept, the change of topology is confined to the sublink equations, therefore the subsystem's matrix need not be recalculated when the change of topology takes place.

3.5 Commentary on Computational Efficiency of the Multilevel MATE Partitioning

The MATE partitioning and its computational advantages are demonstrated in several papers related to the real-time performance of the OVNI simulator [3], [2]. In this work, the advantages of the original MATE concept are extended to the level of a subsystem, bringing along the associated efficiency and flexibility of representation. The application of the newly developed Multilevel MATE concept, therefore, results in further computational speedup through subsystem partitioning and/or more efficient and convenient component modelling. The analytical bases of the proposed approach are derived here, but the extent of its applications reaches far beyond the scope of this thesis work. For example, the approach could allow components of different nature requiring different integration step sizes and/or different integration methods to be combined in the same subsystem. It could also allow the integration of phasor and phase-domain modelling. Exploiting these aspects would result in even more efficient algorithms for the solution of large power system networks.

3.6 Solution of Nonlinearities within the Multilevel MATE Concept

The power system network is nonlinear by nature, and modern simulation tools must allow for accurate modelling of its nonlinear behavior. Nonlinearities are associated with certain power system components; for example, an under-load tap changing (ULTC) transformer will introduce nonlinear behavior with its changing turns ratio. The iron core an electric machine is subject to saturation, which makes the input/output characteristic of the machine nonlinear. In controllers, the limiters, the function of multiplication, etc., will also result in nonlinear system behavior.

Nonlinearities in general can either affect the coefficients of the system equations
(e.g., machine saturation) or the state variables (e.g., nonlinear controllers). Nonlinear elements that cause the coefficients of difference equations to change with time are simpler to understand and implement. An ULTC transformer changes its turns ratio depending on the measured voltage on its low-voltage terminals, which is reflected in the change of coefficients in the system’s matrix. The change of a system’s coefficients with time is referred to as the system’s “changing nature”.

Another type of nonlinearity is described by nonlinear functions applied to state variables of the system, for example, the controller’s sine function, or the multiplier. The nonlinear torque equation in the modelling of a synchronous or induction machine also belongs to this type of nonlinearity. The simultaneous solution of the system’s equations and the nonlinearities that operate on state variables requires an iterative procedure.

The basic principle for solving nonlinearities requiring iterations within the Multilevel MATE framework is explained in general terms in the small example of (3.14). An example of a physical system that reflect this configuration is given at the end of this chapter. In the context of Multilevel MATE, nonlinearities iterate the solution against the link equations alone. For the general derivation of the equations it is sufficient to consider only one nonlinear subsystem, as in (3.14).

\[
\begin{bmatrix}
A & p_A & p_{A,nonlin} \\
-p_A & -z_A & 0 \\
p_{A,nonlin} & 0 & -z_{A,nonlin}
\end{bmatrix} \begin{bmatrix}
p \\
p_0 \\
B
\end{bmatrix} = \begin{bmatrix}
v_A \\
i_{A,sublink} \\
i_{A,nonlin}
\end{bmatrix} = \begin{bmatrix}
h_A \\
-V_{A,sublink} \\
-h_{A,nonlin}
\end{bmatrix}
\]

The nonlinear subsystem \( A \) equations are composed of:

- nodal equations described by the conductance matrix \([A]\), the nodal voltages \([v_A]\), and the vector of accumulated currents \([h_A]\)
- branch sublink equations described by the connectivity arrays \([p_A]\) and \([p'_{A}]\), the sublink Thévenin impedance matrix \([z_A]\), the sublink currents \([i_{A,sublink}]\) and the sublink Thévenin voltages vector \([V_{A,sublink}]\)
- branch nonlinear equations described by the connectivity arrays \([p_{A,nonlin}]\) and \([p'_{A,nonlin}]\), matrix \([z_{A,nonlin}]\), the vector of subsystem’s nonlinear currents \([i_{A,nonlin}]\) and the nonlinear voltage vector \([V_{A,nonlin}]\)

The meaning of the nonlinear equations’ matrices and vectors will be clarified in the example at the end of this section. Here we apply Multilevel MATE to subsystem \( A \) and extract a sublink equation similar to (3.5), now containing a term for the
nonlinear currents \([i_{A,nonlin}]\):
\[
(p_A^t a_A + z_A) \cdot i_{A,sublink} + (p_{A,nonlin}^t a_{A,nonlin}) \cdot i_{A,nonlin} + p_A^t a \cdot i_\alpha = p_A^t e_A + V_{A,sublink}
\]

In a procedure equivalent to that described in Section 3.3, the sublink currents are reduced from the subsystem's equations in (3.14) and the following system of equations is obtained:
\[
\begin{bmatrix}
    1 & a_{MTEmonlin} \\
    0 & p_{A,nonlin}^t a_{MTEmonlin} + z_{A,nonlin}
\end{bmatrix}
\begin{bmatrix}
    a_{MTEmonlin} \\
    0
\end{bmatrix}
\begin{bmatrix}
    v_A \\
    i_{A,nonlin}
\end{bmatrix}
\begin{bmatrix}
    v_B \\
    i_\alpha
\end{bmatrix}
\]

\[
= \begin{bmatrix}
    e_{A,MTEmonlin} \\
    p_{A,nonlin}^t e_{A,MTEmonlin} + V_{A,nonlin}
\end{bmatrix}
\begin{bmatrix}
    v_A \\
    i_{A,nonlin}
\end{bmatrix}
\begin{bmatrix}
    v_B \\
    i_\alpha
\end{bmatrix}
\]

where
\[
a_A = A^{-1} p_A \\
a_{A,nonlin} = A^{-1} p_{A,nonlin} \\
a = A^{-1} p \\
e_A = A^{-1} e_A
\]
\[
a_{MTEmonlin} = a_{A,nonlin} - \Delta a_{A,nonlin} \\
a_{MTEmonlin} = a - e_A \\
e_{A,MTEmonlin} = e_A - \Delta e_A
\]

The nonlinear currents equation is extracted from (3.16) in the following general form:
\[
(p_{A,nonlin}^t \cdot a_{MTEmonlin} + z_{A,nonlin}) \cdot i_{A,nonlin} + p_{A,nonlin}^t \cdot a_{MTEmonlin} \cdot i_\alpha = p_{A,nonlin}^t \cdot e_{A,MTEmonlin} + V_{A,nonlin}
\]

The system of equations is then reduced by moving the unknown nonlinear currents to the right-hand side to obtain the following:
\[
\begin{bmatrix}
    1 & a_{MTEmonlin} \\
    0 & b \\
    0 & p_{A,nonlin}^t a_{MTEmonlin} + q^t b + z
\end{bmatrix}
\begin{bmatrix}
    v_A \\
    v_B \\
    i_\alpha
\end{bmatrix}
\begin{bmatrix}
    e_{A,MTEmonlin} - e_{A,nonlin} \\
    e_B \\
    q^t e_B + V_A
\end{bmatrix}
\]

27
where

$$e_{A,\text{nonlin}} = e_{A,\text{nonlin}} \cdot i_{A,\text{nonlin}}$$

From (3.19) we notice that the nonlinearities contribute only to the subsystem's Thévenin voltage vector. At each solution time step subsystem $A$ assumes the last known value of vector $[i_{A,\text{nonlin}}]$ and calculates the subsystem's Thévenin voltage $[e_{A,\text{MTE}} - e_{A,\text{nonlin}}]$. The link currents are calculated from:

$$i_{\alpha} = z_{\alpha,\text{MTE}}^{-1} (e_{\alpha,\text{MTE}} - e_{\alpha,\text{nonlin}})$$

where

$$e_{\alpha,\text{nonlin}} = p^t \cdot e_{\alpha,\text{nonlin}}$$

and $[z_{\alpha,\text{MTE}}]$ and $[e_{\alpha,\text{MTE}}]$ are as in (3.10). The link currents $[i_{\alpha}]$ are returned to subsystem $A$ and the nonlinear equations are recalculated to produce a new vector $[i_{A,\text{nonlin}}]$. Iterations are repeated until the nonlinear equations satisfy their solution tolerance, then the simulation proceeds to the next time step. The nonlinear equations iterate against the links system alone, and the overall solution for nonlinearities is integrated at the links level. An example of a nonlinear function implementation is shown in the following subsection.

### 3.6.1 Example: A Nonlinear Control Function Implementation with Multilevel MATE

The simultaneous solution of a nonlinear control function and the network requires an iterative procedure. The Multilevel MATE approach offers a computationally efficient framework for the iterative solution of nonlinear equations and is applicable to real-time simulations. We now examine the system in Fig. 3.6 to demonstrate the basic principles of solving a nonlinear control function equation simultaneously with the network solution.

Multiplication of two state variables, for example, makes the control scheme in this example nonlinear. The system in question is partitioned into two subsystems, $A$ and $B$, with subsystem $A$ containing a nonlinear control equation that controls the voltage of node 3. The partitioned system equations can be written in general terms from (3.14) as:

$$\begin{bmatrix}
A & p_{\text{ctrl}} \\
p^t_{\text{ctrl}} & -z_{\text{ctrl}}
\end{bmatrix}
\begin{bmatrix}
p \\
0
\end{bmatrix}
+ \begin{bmatrix}
v_A \\
v_B
\end{bmatrix}
= \begin{bmatrix}
h_A \\
h_B
\end{bmatrix}
+ \begin{bmatrix}
-V_{\text{ctrl}} \\
-V_{\alpha}
\end{bmatrix}$$

where $V_{\text{ctrl}} = v_1 \cdot v_2$ is a nonlinear control function that corresponds to the vector $V_{A,\text{nonlin}}$ in (3.14). Similarly, $[p_{\text{ctrl}}]$, $[p^t_{\text{ctrl}}]$ and $[z_{\text{ctrl}}]$ correspond, using more general
notation, to \([p_{A,nomlin}], [p'_{A,nomlin}]\) and \([z_{A,nomlin}]\), respectively.

For the example in Fig. 3.6, the hybrid system of nodal and branch equations can be written as:

\[
\begin{bmatrix}
3 & -2 & 0 & 0 \\
-2 & 5 & -2 & 0 \\
0 & -2 & 3 & -1 \\
0 & 0 & -1 & 0
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
1 \\
1
\end{bmatrix}
\]

The asterisk (*) sign denotes variables that are recalculated during the iteration process. The nonlinear equation relating the link currents with the nonlinear controller function is extracted as in (3.18):

\[
(p_{ctrl}^t \cdot a_{ctrl} + z_{ctrl}) \cdot i_{ctrl} + p_{ctrl}^t \cdot a \cdot i_\alpha = p_{ctrl}^t \cdot e_A + V_{ctrl}
\]

where \(a_{ctrl} = A^{-1} \cdot p_{ctrl}\). For the example system, the nonlinear equation is:

\[
0.5238 \cdot i_{ctrl} - 0.5238 \cdot i_{link} = -0.1905 + v_1 \cdot v_2
\]

The remaining linear part of the system is represented by the MATE matrix in
Chapter 3. Multilevel MATE

general terms as:

\[
\begin{bmatrix}
1 & 0 & a \\
0 & 1 & b \\
0 & 0 & p^t a + q^t b + z
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
i_\alpha
\end{bmatrix}
= \begin{bmatrix}
e_A - e_{ctrl} \\
e_B \\
p^t (e_A - e_{ctrl}) + q^t e_B + V_A
\end{bmatrix}
\]  \hspace{1cm} (3.23)

where \( e_{ctrl} = a_{ctrl} \cdot i_{ctrl} \) is the contribution of the nonlinear control function to the Thévenin equivalent voltage of subsystem \( A \). For our example, the system of linear equations is:

\[
\begin{bmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
0.1905 \\
0.2857 \\
0.5238
\end{bmatrix}
= \begin{bmatrix}
0.5238 + 0.1905 \cdot i_{ctrl} \\
0.2857 + 0.2857 \cdot i_{ctrl} \\
0.1905 + 0.5238 \cdot i_{ctrl}
\end{bmatrix}
\]

Equations (3.22) and (3.23) completely describe the example system of (3.21) with the clear separation between nonlinear and linear parts, both represented by their Thévenin equivalents as seen from the system's linking nodes 3 and 4. To recapitulate, the two equations that iterate against each other at each time step of the simulation are in general terms:

\[
(p^t_{ctrl} \cdot a_{ctrl} + z_{ctrl}) \cdot i_{ctrl} = p^t_{ctrl} \cdot e_A + V_{ctrl} - p^t_{ctrl} \cdot a \cdot i_\alpha \]  \hspace{1cm} (3.24)

\[
z_\alpha \cdot i_\alpha = e_\alpha - e_{a.ctrl} \]  \hspace{1cm} (3.25)

where, for this case, \( e_{a.ctrl} = p^t \cdot a_{ctrl} \cdot i_{ctrl} \), or evaluated:

\[
0.5238 \cdot i_{ctrl} = -0.1905 + v_1 \cdot v_2 + 0.5238 \cdot i_{link}
\]

\[
3.0238 \cdot i_{link} = -0.8095 + 0.5238 \cdot i_{ctrl}
\]

At each iteration step, subsystem \( A \) assumes the last known value of the nonlinear controller current \( i_{ctrl} \) and calculates its contribution to \( e_{a.ctrl} \), which modifies the Thévenin equivalent voltage. The modified Thévenin equivalent of subsystem \( A \) is then passed to the links to obtain the link current from (3.25). The calculated link current is returned to subsystem \( A \). The nonlinear controller requests the value of the link current \( (i_\alpha) \) from the subsystem as well as the voltages \( v_1 \) and \( v_2 \) needed to recalculate its nonlinear equation (3.24). The new value of \( i_{ctrl} \) is passed back to subsystem \( A \) in order to recalculate its modified Thévenin equivalent voltage. The modified Thévenin equivalent voltage of subsystem \( A \) is again passed to the links, and the new value of the link current is obtained. Iterations are repeated until the
Chapter 3. Multilevel MATE

nonlinear controller satisfies its solution tolerance and raises a flag for the simulation to proceed to the next time step. The iteration procedure is depicted as a flow chart in Fig. 3.7. For our example, the nonlinear iterations produce the following output:

<table>
<thead>
<tr>
<th>i</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>7</th>
<th>8</th>
<th>9</th>
<th>10</th>
<th>11</th>
<th>12</th>
<th>13</th>
<th>14</th>
</tr>
</thead>
<tbody>
<tr>
<td>$i_m$</td>
<td>-0.2677</td>
<td>-0.3082</td>
<td>-0.3295</td>
<td>-0.3453</td>
<td>-0.3478</td>
<td>-0.3489</td>
<td>-0.3495</td>
<td>-0.3498</td>
<td>-0.3499</td>
<td>-0.3500</td>
<td>-0.3500</td>
<td>-0.3500</td>
<td>-0.3500</td>
<td>-0.3500</td>
</tr>
<tr>
<td>$v_i$</td>
<td>0.5748</td>
<td>0.5380</td>
<td>0.5186</td>
<td>0.5090</td>
<td>0.5043</td>
<td>0.5020</td>
<td>0.5009</td>
<td>0.5004</td>
<td>0.5002</td>
<td>0.5001</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
<td>0.5000</td>
</tr>
<tr>
<td>$v_o$</td>
<td>0.3622</td>
<td>0.3069</td>
<td>0.2779</td>
<td>0.2634</td>
<td>0.2564</td>
<td>0.2530</td>
<td>0.2514</td>
<td>0.2506</td>
<td>0.2503</td>
<td>0.2501</td>
<td>0.2500</td>
<td>0.2500</td>
<td>0.2500</td>
<td>0.2500</td>
</tr>
<tr>
<td>$l_m$</td>
<td>-0.2340</td>
<td>-0.3567</td>
<td>-0.4180</td>
<td>-0.4478</td>
<td>-0.4621</td>
<td>-0.4690</td>
<td>-0.4722</td>
<td>-0.4737</td>
<td>-0.4745</td>
<td>-0.4748</td>
<td>-0.4750</td>
<td>-0.4750</td>
<td>-0.4751</td>
<td>-0.4751</td>
</tr>
</tbody>
</table>

Table 3.2: Iterations for the nonlinear controller equation

The nonlinear controller iterates the solution for its current against the link equations alone. If there were more nonlinear controllers in the system, each one would perform its iterating procedure in the same manner, with the overall solution for nonlinearities being integrated at the links level. For slow-changing control variables in power system networks (e.g., RMS values of system voltages and currents), the algorithm automatically recognizes that the iteration procedure is not needed. In other words, the links solution is based on the present time step of the subsystems’ Thévenin equivalents and the previous time step of the nonlinear currents. The values of the present time step nonlinear controller currents are calculated from the nonlinear equations and are used to obtain the present time step nodal subsystem voltages.

The iterative procedure that we used for solving nonlinearities corresponds to the fixed point iteration method. The main reason we chose this method is its simplicity and generality. The fixed point iteration method requires less computation time at each iteration step than the more complex Newton type iteration methods. Also, because it does not require calculation of the Jacobian matrix, the method is more general for the implementation of user-built nonlinear controller equations. It should be noted that even though the convergence of the Newton type methods could be faster, the applicability of these methods to solving nonlinear systems within the Multilevel MATE concept needs to be investigated in the future.

### 3.7 Multilevel MATE Block Diagram

Fig. 3.8 shows a block diagram of the Multilevel MATE implementation. The diagram describes the computational requirements and the flow of information from the lowest level (single element) to the highest level (subsystem’s Thévenin equivalent and the links).
3.8 Summary

This chapter introduces the new concept of Multilevel MATE, which is suitable for implementation in the real-time simulator OVNI. The advantages of subsystem partitioning are presented in terms of computational speedup, and the advantages of modelling with branch equations are demonstrated through a switch implementation and a nonlinear control function implementation.
Chapter 3. Multilevel MATE

Figure 3.8: Block diagram of Multilevel MATE implementation

33
Chapter 4

Modelling with Multilevel MATE

4.1 Introduction

The Multilevel MATE concept offers many advantages in the modelling of power system networks and their components, as will be shown in this chapter. Some of the advantages are associated with higher computational efficiency of the models compared with other current modelling techniques. An example demonstrating this efficiency is the model of a switch described in the preceding chapter. Multilevel MATE also provides a convenient and very general approach to the modelling of ideal voltage sources, whether grounded or ungrounded, by the means of sublinks.

Other advantages of Multilevel MATE are associated with modelling accuracy. This is especially evident in cases of modelling of ungrounded or weakly grounded circuits and subsystems. The nodal analysis typically used in the EMTP type of solution may suffer ill-conditioning problems when weak connections or no connections to the common reference point (ground) exist. This is reflected in the system’s admittance matrix becoming singular. A similar problem arises in the modelling of networks that are grounded only through voltage sources in the context of the MATE solution framework in the OVNI simulator. In this chapter, we present a new approach to achieving good conditioning even under these limiting situations. It solves the problem of ill-conditioning for many power system components.

Finally, we present the Multilevel MATE-based modelling of power system components used in OVNI, which offers a simple, more general and more intuitive modelling approach.

The research contributions reported in this chapter include the following:

- Description of a new technique for conditioning of ungrounded or weakly grounded circuits and subsystems.
- Implementation of power system components such as ideal voltage sources, phase-domain induction and synchronous machines, and controllers with functional sublinks.
4.2 Modelling of Ideal Voltage Sources

Within the concept of Multilevel MATE, ideal voltage sources are naturally modelled as functional sublinks. This approach has several advantages, the most important one being generality. Whether an ideal voltage source is connected to the ground or between the nodes in the circuit, its equation is constructed and solved in exactly the same manner. Also, the choice of calculating the voltage source current (and therefore the power supplied to the network) is inherently available and remains with the user. A simple example to demonstrate ideal voltage source modelling is shown in Fig. 4.1.

![Figure 4.1: An example system to demonstrate modelling of ideal voltage sources](image)

The circuit contains two voltage sources, grounded \( V_1 \) connected between node "1" and ground and ungrounded \( V_2 \) connected between nodes "2" and "3". The system of equations with voltage sources modelled as sublinks can be written as

\[
\begin{bmatrix}
g_{12} & -g_{12} & 0 & 0 & -1 & 0 \\
-g_{12} & g_{12} + g_{20} & 0 & 0 & 0 & -1 \\
0 & 0 & g_{34} & -g_{34} & 0 & +1 \\
0 & 0 & -g_{34} & g_{34} + g_{40} & 0 & 0 \\
-1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & +1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4 \\
i_1 \\
i_2
\end{bmatrix}
= \begin{bmatrix}
0 \\
v_2 \\
v_3 \\
v_4 \\
-V_1 \\
-V_2
\end{bmatrix}
\]  

(4.1)

where (4.1) corresponds to the MATE subsystem matrices and vectors described as

\[
\begin{bmatrix}
A & p_A \\
p^T_A & -z_A
\end{bmatrix}
\begin{bmatrix}
v_A \\
i_{A\text{sublink}}
\end{bmatrix}
= \begin{bmatrix}
h_A \\
-V_{A\text{sublink}}
\end{bmatrix}
\]  

(4.2)

Referring to the procedure of solving the system of hybrid equations using the Multilevel MATE concept described in Section 3.3, the solution for the system's
nodal voltages and ideal voltage source currents can be found. Voltage sources will contribute to the Thévenin equivalent voltage vector $e_{A,MTE}$ of the subsystem to obtain:

$$i_{A,\text{sublink}} = (p_A^t A^{-1} p_A + z_A)^{-1} \cdot (p_A^t A^{-1} h_A + V_{A,\text{sublink}}) = \begin{bmatrix} 1.2727 & 0.1818 \end{bmatrix}^T$$

$$v_A = e_{A,MTE} = A^{-1} p_A \cdot i_{A,\text{sublink}} = \begin{bmatrix} 2.0000 & 0.7273 & -0.2727 & -0.0909 \end{bmatrix}^T$$

### 4.2.1 Commentary on Modelling of Ideal Voltage Sources

By examining the hybrid system of equations (4.1) one can observe that in the traditional EMTP formulation, the system of nodal equations would be smaller. The reason for this is that in the EMTP ideal voltage sources are internally converted into Norton equivalents. To be more specific, an ideal voltage source would "borrow" the series impedance from the network, converting it in essence into its internal source impedance. In OVNI, we rely on object-oriented design, where the identity of each component must be preserved. Therefore, the internal impedance of a voltage source is its property and does not depend on the connected network.

Regardless of whether a voltage source has an internal impedance or not, it will contribute to the hybrid system of equations with a branch equation (functional sublink). If a source is non-ideal, the internal nodal voltage (e.g., node "1" in Fig. 4.1) would not be calculated, which corresponds to the situation in the EMTP (there is no increase in the number of nodal equations over the EMTP approach). If a voltage source is ideal, its internal impedance will be set to zero and the system of equations will have the form as in (4.1). In both cases, the calculation of the source current is optional. To recapitulate, the user has the flexibility of specifying a voltage source as ideal, in which case its nodal voltage would be included in the solution, or non-ideal, in which case its internal node would be non-existent to the surrounding network, resulting in a smaller system of nodal equations. This approach is more general than that implemented in the EMTP and is especially useful in OVNI when, for example, a subsystem (at any level of partitioning) consists of voltage sources only (ideal or non-ideal), that simply represent the Thévenin equivalent of the corresponding subsystem.

### 4.3 Modelling of Ungrounded Circuits and Subsystems

Modelling of ungrounded circuits and subsystems using nodal equations results in singularity of their conductance matrices in OVNI. To overcome this problem, different solution alternatives are presented for ungrounded circuits and ungrounded subsystems. In this thesis the term ungrounded circuit refers to a circuit configura-
tion where the reference point connection (ground) does not exist. An example of such a configuration is an ungrounded secondary side of a power system transformer. The term ungrounded subsystem refers to a subsystem network that is connected to the ground through links only. In other words, the reference point is unknown when solving for a subsystem's network, but becomes known once the solution for the links is found. An example of this configuration would be, for example, an ungrounded three phase induction machine connected through links to a grounded network. Subsystems and circuits are also considered to be ungrounded if grounded through ideal voltage sources only.

4.3.1 Ungrounded Circuits

To demonstrate the proposed solution for modelling ungrounded circuits, a simple network configuration containing a single-phase ideal transformer is shown in Fig. 4.2. The system has four nodes, two of which ("3" and "4") belong to the ungrounded circuit.

\[ I_1 = 2 \cos(\omega t) \]

Figure 4.2: An example system to demonstrate modelling of ungrounded circuits

The ideal transformer model introduces a branch equation (functional sublink) describing either one of the currents \( i_p \) or \( i_s \) that are related to each other with the transformer's turns ratio \( n = i_p / i_s \). The hybrid system of equations for this particular case can be written in general terms as:

\[
\begin{bmatrix}
 g_{10} + g_{12} & -g_{12} & 0 & 0 & 0 \\
 -g_{12} & g_{12} & 0 & 0 & n \\
 0 & 0 & g_{34} & -g_{34} & -1 \\
 0 & 0 & -g_{34} & g_{34} & +1 \\
 0 & n & -1 & +1 & 0
\end{bmatrix}
\begin{bmatrix}
 v_1 \\
 v_2 \\
 v_3 \\
 v_4 \\
 i_s
\end{bmatrix}
=
\begin{bmatrix}
 J_1 \\
 0 \\
 0 \\
 0 \\
 0
\end{bmatrix}
\]  

(4.3)

The system’s conductance matrix in (4.3) is singular, and therefore the system cannot be solved using the MATE concept. Let us first comment on the cause of
the matrix singularity. Nodal voltages \(v_3\) and \(v_4\) can only be calculated relative to a common reference point. As the common reference point does not exist on the secondary side of the transformer, there are an infinite number of solutions for these two voltages that satisfy the ideal transformer relationship \(v_3 - v_4 = n \cdot v_2\). Unless the common reference point is introduced artificially, the solution for nodal voltages cannot be obtained.

Now when the problem and its solution have been identified, the common reference point needs to be introduced without affecting the solution of the system. One approach is to ground one of the nodes (e.g., node "4") on the transformer's secondary side. In this way the voltage \(v_4\) becomes known and the system of nodal equations has a unique solution. This approach, however, affects the system's solution in the case of unbalanced network conditions (e.g., line-to-ground fault) when simulating the operation of a three-phase transformer, by allowing the flow of a zero sequence current through the ground. To maintain the approach general enough for any number of transformer windings and winding connections, the approach shown in Fig. 4.3 is proposed.

![Figure 4.3: Modelling of an ungrounded transformer circuit with the Multilevel MATE approach](image)

Shunt impedances \(z_{cs} = 1\Omega\) can be inserted from nodes "3" and "4" to the common reference point, in this case chosen to be the ground potential of 0 V. To compensate for the extra current that the branch voltage \(v_s = v_3 - v_4\) has to supply to the shunt impedances, voltage-dependent current sources of \(v_s/2\) are added to the circuit at nodes "3" and "4". In this work, the shunt impedances, whose purpose is to introduce a common reference point in ungrounded circuits, are referred to as
the compensating shunt impedances. The circuit of Fig. 4.3 can also be depicted in a more compact representation as shown in Fig. 4.4.

![Circuit Diagram](image)

Figure 4.4: Introducing a common reference point to the ungrounded secondary side of an ideal transformer

The system of equations describing the secondary side network configuration shown in Fig. 4.3 or Fig. 4.4 can be written as

\[
\begin{bmatrix}
  g_{10} + g_{12} & -g_{12} & 0 & 0 & 0 \\
  -g_{12} & g_{12} & 0 & 0 & n \\
  0 & -n/2 & g_{34} + y_{csh} & -g_{34} & -1 \\
  0 & n/2 & -g_{34} & g_{34} + y_{csh} & +1 \\
  0 & n & -1 & +1 & 0
\end{bmatrix}
\begin{bmatrix}
  v_1 \\
  v_2 \\
  v_3 \\
  v_4 \\
  i_s
\end{bmatrix}
= 
\begin{bmatrix}
  I_1 \\
  0 \\
  0 \\
  0 \\
  0
\end{bmatrix}
\]

(4.4)

At \( t = 0 \) of the simulation, the circuit of Fig. 4.3 with the artificially introduced reference point on the secondary side of the transformer will produce the following solution for its currents and voltages:

\[
i_s = i_{A,sublink} = (p_A^t \cdot a_A + z_A)^{-1} \cdot (p_A^t \cdot e_A + V_{A,sublink}) = 0.13245 \ A
\]

\[
\begin{bmatrix}
  v_1 \\
  v_2 \\
  v_3 \\
  v_4
\end{bmatrix}
= 
\begin{bmatrix}
  0.67550 \\
  0.013245 \\
  0.066225 \\
  -0.066255
\end{bmatrix}
\]

where (4.4) corresponds to the MATE matrices and vectors described in (4.2) and

\[
a_A = A^{-1} p_A \\
e_A = A^{-1} h_A
\]
Chapter 4. Modelling with Multilevel MATE

Note that with this approach, the nodal solution can be obtained even when the secondary side of the transformer is unloaded, by setting the conductance between nodes "3" and "4" to zero ($g_{34} = 0$) and changing the sublink equation to $i_s = 0$ in (4.4).

The correctness of the solution can be checked against that obtained using the branch voltage equation $v_s \cdot g_{34} - i_s = 0$, instead of the two equations for nodal voltages $v_3$ and $v_4$, in the secondary transformer circuit. Equation (4.3) then becomes:

$$\begin{bmatrix}
g_{10} + g_{12} & -g_{12} & 0 & 0 \\
-g_{12} & g_{12} & 0 & n \\
0 & 0 & g_{34} & -1 \\
0 & n & -1 & 0
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_s \\
i_s
\end{bmatrix}
=
\begin{bmatrix}
I_1 \\
0 \\
0 \\
0
\end{bmatrix}
$$

with the solution

$$i_s = 0.1325 \ \text{A}$$

$$\begin{bmatrix}
v_1 \\
v_2 \\
v_s
\end{bmatrix}^T = \begin{bmatrix}
0.6755 & 0.0132 & 0.1325
\end{bmatrix}^T V$$

4.3.1.1 Ungrounded Ideal Voltage Sources in Ungrounded Circuits

In a more general case, an ungrounded circuit that contains ungrounded (branch) ideal voltage sources can be artificially grounded in the same manner as shown in the case of the ungrounded transformer circuit in Fig. 4.4. However, there can be only one artificially introduced reference point in the ungrounded circuit, meaning that the compensating shunt impedances can be added to only one ungrounded voltage source. To demonstrate this, a simple network with two ungrounded ideal voltage sources is considered in Fig. 4.5.

![Network Diagram](image)

Figure 4.5: Introducing a common reference point to the ungrounded network with two ungrounded ideal voltage sources

By adding no compensating shunt impedances (shown dashed in Fig. 4.5) to the ungrounded voltage source $V_1$, the system's conductance matrix is non-invertible and

40
Chapter 4. Modelling with Multilevel MATE

The solution for nodal voltages cannot be found. By introducing the common reference point, the system's equations become

\[
\begin{bmatrix}
g_{12} + y_{csh} & -g_{12} & 0 & 0 & 0 & -1 & 0 \\
-g_{12} & g_{12} + g_{23} & -g_{23} & 0 & 0 & 0 & 0 \\
0 & -g_{23} & g_{23} & 0 & 0 & 0 & 0 \\
0 & 0 & g_{45} + y_{csh} & -g_{45} & 0 & +1 & 0 \\
0 & 0 & 0 & -g_{45} & g_{45} + g_{56} & -g_{56} & 0 & 0 \\
0 & 0 & 0 & 0 & -g_{56} & g_{56} & 0 & +1 \\
-1 & 0 & 0 & +1 & 0 & 0 & 0 & 0 \\
0 & 0 & -1 & 0 & 0 & +1 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4 \\
v_5 \\
v_6 \\
i_1 \\
i_3
\end{bmatrix}
= \begin{bmatrix}
v_4/2 \\
0 \\
0 \\
-\sqrt{2} \\
0 \\
0 \\
-V_1 \\
-V_3
\end{bmatrix}
\]

and can now be solved, with the following solution:

\[
\begin{bmatrix}
i_1 & i_3\end{bmatrix}^T = \begin{bmatrix}
-2.4 & 2.4
\end{bmatrix}^T A
\]

\[
\begin{bmatrix}
v_1 & v_2 & v_3 & v_4 & v_5 & v_6\end{bmatrix}^T = \begin{bmatrix}
2.5 & 4.9 & 6.1 & -2.5 & -3.3 & -3.9
\end{bmatrix}^T V
\]

The current flowing through the circuit \( i_1 = -i_3 \) can also be calculated from Kirchhoff's Voltage Law to verify the solution of the ungrounded network

\[
i_1 = \frac{V_1 - V_3}{g_{23} + g_{34} + g_{45} + g_{56}} = -2.4 \quad A
\]

If a second reference point were introduced through the compensating shunt impedances at voltage source \( V_3 \), the obtained system solution would be incorrect. For the example system it would give the solution \( i_1 = -2.5704 \) A.

4.3.1.2 Grounded Ideal Voltage Sources in Ungrounded Circuits

Let us now consider the system in Fig. 4.6, which has two grounded ideal voltage sources. The nodal conductance matrix of the system is singular, unless a shunt impedance is introduced. With a similar approach as for the ungrounded voltage sources previously discussed, the shunt impedance is inserted at the voltage source node, and the extra current supplied to the newly added shunt element is compensated for by a voltage-dependent current source. Note that in the case with grounded voltage sources, the circuit is anchored to a fixed reference point. Therefore, inserting compensating shunt elements at more than one grounded voltage source node will not modify the overall network solution (but is, of course, unnecessary).

41
Figure 4.6: Introducing a common reference point to the ungrounded network with two grounded ideal voltage sources

For the system in Fig. 4.6, the hybrid system of equations can be written as

\[
\begin{bmatrix}
g_{12} + y_{csh} & -g_{12} & 0 & -1 & 0 \\
-g_{12} & g_{12} + g_{23} & -g_{23} & 0 & 0 \\
0 & -g_{23} & g_{23} + y_{csh} & 0 & -1 \\
-1 & 0 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
i_1 \\
i_3 \\
\end{bmatrix}
= \begin{bmatrix}
V_1 \\
0 \\
V_3 \\
-V_1 \\
-V_3 \\
\end{bmatrix}
\]

with the solution

\[
\begin{bmatrix}
i_1 \\
i_3 \\
v_1 \\
v_2 \\
v_3 \\
\end{bmatrix}^T = \begin{bmatrix}
-3.3333 & 3.3333 \\
5.0 & 8.3 & 10.0 \\
\end{bmatrix}^T V
\]

4.3.2 Ungrounded Subsystems

In the conventional MATE approach, if a subsystem is not connected to a common reference point (ground), its conductance matrix cannot be inverted, and therefore the Thévenin equivalent of the subsystem cannot be found. Unlike the case of ungrounded networks, the connection to the reference point does exist outside of the ungrounded subsystem, and there is a unique solution for the network's nodal voltages.

Ungrounded subsystems can also be thought of as subsystems with a reference point that is different from the ground. The nodal solution for the subsystem voltages can only be obtained if the reference voltage can somehow be calculated prior to obtaining the solution for its nodal voltages. With this in mind, we derive the following example to demonstrate how ungrounded subsystems can be handled in OVNI. The ungrounded wye-connected three-phase resistive load in Fig. 4.7 represents a good example of an ungrounded subsystem connected to a grounded network. The voltage sources on the grounded side of the network are chosen so that the system is perfectly balanced. The simple configuration of Subsystem B can be considered as the Thévenin equivalent of a more complex subsystem, as seen from the linking
nodes. Let us now find the Thévenin equivalent of the ungrounded subsystem A. The Thévenin equivalent of any ungrounded subsystem as seen from the linking nodes (in this case nodes 1, 2 and 3) can be found by choosing any one of the nodes to be a subsystem’s reference point. In the example, we arbitrarily choose node 3 to be the reference point for the ungrounded subsystem. Subsystem A’s equations are derived with respect to the new reference point $v_{3ref}$ as follows:

$$
\begin{bmatrix}
g_{14} & 0 & -g_{14} \\
0 & g_{24} & -g_{24} \\
-g_{14} & -g_{24} & g_{14} + g_{24} + g_{34}
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_4
\end{bmatrix}
+ 
\begin{bmatrix}
-1 & 0 & 0 \\
0 & -1 & 0 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
i_{a_{link}} \\
i_{b_{link}} \\
i_{c_{link}}
\end{bmatrix}
= 
\begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
- 
\begin{bmatrix}
0 \\
0 \\
-g_{34}
\end{bmatrix}
\cdot v_{3ref}
$$

This equation can be written in more general terms as:

$$
A \cdot v_A + p \cdot i_\alpha = h_A - h_{A,ref}
$$

using the familiar notation for MATE partitioning, and introducing the new term $h_{A,ref}$ that corresponds to the contribution of the reference voltage to the subsystem’s vector of accumulated currents:

$$
h_{A,ref} = p_{A,vref} \cdot v_{ref}
$$

By multiplying the system of equations with $A^{-1}$, we obtain the following expression:

$$
v_A + a \cdot i_\alpha = e_A - 
\begin{bmatrix}
-1 \\
-1 \\
-1
\end{bmatrix}
\cdot v_{ref}
$$

(4.6)
Chapter 4. Modelling with Multilevel MATE

From (4.6) we see that the reference voltage increases all nodal voltages in the ungrounded subsystem by the same amount. The nodal voltages in subsystem A are therefore referenced to the voltage of node 3. Therefore, the Thévenin equivalent as seen from the linking nodes 1, 2 and 3 is:

\[
z_{\alpha} = p^t a + q^t b + z = \begin{bmatrix} 1 & 0.5 & 0 \\ 0.5 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \begin{bmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 2.5 & 0.5 & 0 \\ 0.5 & 2.5 & 0 \\ 0 & 0 & 1.5 \end{bmatrix}
\]

\[
e_{\alpha} = p^t e_A + q^t e_B + V_{\alpha} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 2 \\ -1 \\ 0 \end{bmatrix}
\]

\[(4.7)\]

The reference voltage of subsystem A is calculated at the links level and is uniquely defined by the relationship for link currents that is valid for ungrounded subsystems only: the sum of link currents coming into the subsystem has to be equal to the sum of the link currents coming out of the subsystem. If we combine this relationship with the links system of equations representing the system's Thévenin equivalent as seen from the linking nodes, we will obtain, for the example system, the following hybrid system of equations:

\[
\begin{bmatrix}
2.5 & 0.5 & 0 & +1 \\
0.5 & 2.5 & 0 & +1 \\
0 & 0 & 1.5 & +1 \\
+1 & +1 & +1 & 0
\end{bmatrix}
\begin{bmatrix}
i_{\alpha,\text{link}} \\
i_{\beta,\text{link}} \\
i_{\gamma,\text{link}} \\
v_3
\end{bmatrix} = \begin{bmatrix}
2 \\
-1 \\
-1 \\
0
\end{bmatrix}
\]

which can be written, using the more general MATE notation, as:

\[
\begin{bmatrix}
z_{\alpha} \\
p_{v_{\alpha}} \\
z_{v_{v_{\alpha}}}
\end{bmatrix} \cdot \begin{bmatrix}
i_{\alpha} \\
v_{\alpha} \\
v_{v_{\alpha}}
\end{bmatrix} = \begin{bmatrix}
e_{\alpha} \\
h_{v}_{\alpha}
\end{bmatrix}
\]

\[(4.8)\]

By applying the MATE inversion concept to these modified link equations, reference voltage equation can be extracted and solved independently from the link currents. The connectivity array \(p_{v_{\alpha}}\) that resembles the \(p\) and \(q\) connectivity arrays on the subsystem level can be calculated from

\[
p_{v_{\alpha}} = p^t A^{-1} p_{A, v_{\alpha}}\]

and

\[
p_{v_{\alpha}}^t = [p_{v_{\alpha}}]^T\]

\[(4.9)\]

\[(4.10)\]
For completeness, the equation expressions to calculate $z_{v_{\text{ref}}}$ and $h_{v_{\text{ref}}}$ are given as

$$z_{v_{\text{ref}}} = p_{A_{\text{vref}}}^t A^{-1} p_{A_{\text{vref}}} - z_{A_{\text{vref}}},$$

(4.11)

$$h_{v_{\text{ref}}} = p_{A_{\text{vref}}}^t A^{-1} h_{A} - V_{A_{\text{vref}}},$$

(4.12)

where the terms $z_{A_{\text{vref}}}$ and $V_{A_{\text{vref}}}$ are defined by a general expression for subsystem A reference voltages as

$$p_{A_{\text{vref}}}^t v_{A} + z_{A_{\text{vref}}} v_{\text{ref}} = V_{A_{\text{vref}}}.$$  

(4.13)

The reference voltage equation for the ungrounded subsystem can be extracted and solved as follows

$$v_{\text{3ref}} = v_{\text{ref}} = \left( p_{v_{\text{ref}}}^t z_{\alpha}^{-1} p_{v_{\text{ref}}} - z_{v_{\text{ref}}} \right)^{-1} \cdot \left( p_{v_{\text{ref}}}^t z_{\alpha}^{-1} \right. e_{\alpha} - h_{v_{\text{ref}}} \left. \right)$$

$$= \left( \begin{bmatrix} +1 & +1 & +1 \\ 0.5 & 2.5 & 0 \\ 0 & 0 & 1.5 \end{bmatrix} \right)^{-1} \left( \begin{bmatrix} +1 \\ +1 \\ +1 \end{bmatrix} \right) - \left( \begin{bmatrix} 2 \\ -1 \\ -1 \end{bmatrix} \right) = -0.25 \text{ V}$$

(4.14)

The link currents can then be calculated from (4.8), taking into account the contribution of the reference voltage to the Thévenin equivalent as:

$$\begin{bmatrix} i_{a_{\text{link}}} \\ i_{b_{\text{link}}} \\ i_{c_{\text{link}}} \end{bmatrix} = i_{\alpha} = z_{\alpha}^{-1} \cdot \left( e_{\alpha} - p_{v_{\text{ref}}} \cdot v_{\text{ref}} \right) = \begin{bmatrix} 1 \\ -0.5 \\ -0.5 \end{bmatrix}$$

(4.15)

The links system passes the values of the link currents back to subsystems A and B and, in addition, the value of the reference voltage to the ungrounded subsystem A. Finally, the solution for nodal voltages can be calculated for subsystems A and B as:

$$v_{A} = e_{A} - A^{-1} p_{A_{\text{vref}}} \cdot v_{\text{ref}} - a \cdot i_{\alpha} = \begin{bmatrix} 0.5 \\ -0.25 \\ 0 \end{bmatrix} V$$

(4.16)

$$v_{B} = e_{B} - b \cdot i_{\alpha} = \begin{bmatrix} 1.5 \\ -0.75 \\ -0.75 \end{bmatrix} V$$

In a more general approach, the reference node does not need to be reduced from the ungrounded subsystem equations but can be duplicated in order to keep the size
and structure of the system of nodal equations untouched. The effect of duplicating the node appears in the equations as adding a conductance of one siemens between the original node 3 and its reference duplicate \(3_{\text{ref}}\). However, since \(v_3 = v_{3_{\text{ref}}}\) there will be no current flowing between nodes 3 and \(3_{\text{ref}}\), meaning that the newly introduced reference point does not affect the solution of the system. Such a case (4.5) will have the following form instead:

\[
\begin{bmatrix}
g_{14} & 0 & 0 & -g_{14} \\
0 & g_{24} & 0 & -g_{24} \\
0 & 0 & g_{34} + 1 & -g_{34} \\
-g_{14} & -g_{24} & -g_{34} & g_{14} + g_{24} + g_{34}
\end{bmatrix}
\begin{bmatrix}
v_1 \\
v_2 \\
v_3 \\
v_4
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
\]

\[
\begin{bmatrix}
-1 & 0 & 0 \\
0 & -1 & 0 \\
0 & 0 & -1 \\
0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
i_{a,\text{link}} \\
i_{b,\text{link}} \\
i_{c,\text{link}}
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix}
\cdot v_{3_{\text{ref}}}
\] (4.17)

Even though this more general approach is the one adopted for OVNI, the approach with the reduced reference node provides a more intuitive understanding of what, for example, \(p_{\text{ref}}\) represents. The bottom line is that both approaches produce identical systems of equations (4.8).

Note that the choice of reference point for the ungrounded subsystems is arbitrary. In fact, in the particular case examined here, the choice of the neutral point of a three-phase resistor's wye connection (node 4) would have been the more natural choice. Under balanced conditions, as was the case here, the voltage of the ungrounded neutral point is equal to zero. As will be shown in the next section, a model of a three-phase electric machine with ungrounded stator windings falls into the category of ungrounded subsystems.

### 4.4 Modelling of Electric Machines

In this section, we describe the modelling of three-phase induction and synchronous machines suitable for transient analysis with the Multilevel MATE concept. Electric machines are the most complex electric power components and therefore the most difficult to accurately model for detailed simulation analyses. Over the years, certain models have been accepted as a good machine representation for transient stability analysis, as well as for the studies using the electromagnetic transients programs (EMTP) [27],[28].

Both approaches adopt machine modelling in \(dq0\) coordinates, which leads to less demanding computational requirements. The \(dq0\) modelling is especially convenient for the implementation into transient stability programs, where power system net-
works are represented with their single-phase equivalents. The situation when simulating a $dq0$ machine model with the EMTP-type of programs is less favourable, as the machine model has to be interfaced with the three-phase detailed network representation. Also, since the EMTP-type of programs are generally used for a very accurate simulation of locally spread network phenomena, the $dq0$ model is limited in its representation of the internal machine structure (internal faults, saturation effects). With this limitation in mind, phase-domain synchronous and induction machine models were proposed in [25], [29].

The phase-domain models of three-phase induction and synchronous machines of [25], [29] have been implemented in OVNI exploiting the Multilevel MATE concept. The detailed description of the derivation of the phase-domain induction machine model is presented and its implementation with Multilevel MATE is explained. Because of the similarities in modelling of the induction and synchronous machine phase-domain models that will be outlined in the corresponding section, only a brief description of the implementation of the synchronous machine phase-domain model with the Multilevel MATE is given here. The implementations of the induction and synchronous machine phase-domain models were tested and successfully compared against results from available literature.

4.4.1 Phase-Domain Induction Machine Model

4.4.1.1 Electrical Equations of an Induction Machine

Stator and rotor electrical circuits of the induction machine are, in essence, mutually coupled three-phase inductances, where the stator circuit is stationary and the rotor circuit rotates at an angular velocity $\omega_r$ with respect to the stator, as shown in Fig. 4.8. From fundamental circuit theory, the voltages across the windings (phases) of an induction machine can be expressed for the stator as:

$$e_a = \frac{\partial \phi_a}{\partial t} + R_a i_a$$
$$e_b = \frac{\partial \phi_b}{\partial t} + R_b i_b$$
$$e_c = \frac{\partial \phi_c}{\partial t} + R_c i_c$$

and for the rotor as:

$$e_A = \frac{\partial \phi_A}{\partial t} + R_A i_A$$
$$e_B = \frac{\partial \phi_B}{\partial t} + R_B i_B$$
$$e_C = \frac{\partial \phi_C}{\partial t} + R_C i_C$$

---

5 Figure courtesy of P. Kundur: Power System Stability and Control
where the flux linkages ($\psi$) in the stator phase $a$ and the rotor phase $A$ are given by:

$$
\psi_a = L_{aa}i_a + L_{ab}i_b + L_{ac}i_c + L_{Aa}i_A + L_{AB}i_B + L_{AC}i_C \\
\psi_A = L_{AA}i_a + L_{BA}i_b + L_{CA}i_c + L_{AA}i_A + L_{AB}i_B + L_{AC}i_C
$$

(4.20)

This can be written in a matrix form for all stator ($a, b, c$) and rotor ($A, B, C$) phases as:

$$
\begin{bmatrix}
\psi_a \\
\psi_b \\
\psi_c \\
\psi_A \\
\psi_B \\
\psi_C
\end{bmatrix} =
\begin{bmatrix}
L_{aa} & L_{ab} & L_{ac} & L_{Aa} & L_{AB} & L_{AC} \\
L_{ab} & L_{bb} & L_{bc} & L_{BA} & L_{BB} & L_{BC} \\
L_{ac} & L_{bc} & L_{cc} & L_{CA} & L_{CB} & L_{CC} \\
L_{Aa} & L_{BA} & L_{CA} & L_A & L_B & L_C \\
L_{AA} & L_{BA} & L_{CA} & L_{AB} & L_{BB} & L_{BC} \\
L_{AC} & L_{BC} & L_{CC} & L_{AB} & L_{BC} & L_{CC}
\end{bmatrix}
\begin{bmatrix}
i_a \\
i_b \\
i_c \\
i_A \\
i_B \\
i_C
\end{bmatrix}
$$

(4.21)

where

- $L_{is}$ and $L_{ir}$ are leakage inductances of stator and rotor windings respectively
- $L_{ms}$ is a magnetizing inductance of the stator winding
• \( L_{mr} \) is a magnetizing inductance of the rotor winding

• \( L_{sr} \) is the amplitude of the mutual inductance between stator and rotor windings

From (4.21) it can be noted that the mutual inductances between the rotor and stator windings depend on the rotor angular position \( \theta_r(t) \) and are, therefore, time-dependent.

\[
\begin{align*}
\psi_1(t) &= L_{11}(t) \cdot i_1(t) + L_{12}(t) \cdot i_2(t) \\
\psi_2(t) &= L_{12}(t) \cdot i_1(t) + L_{22}(t) \cdot i_2(t)
\end{align*}
\]

can be discretized by applying the trapezoidal implicit integration rule, described in Appendix A, to obtain the following system of equations:

\[
\begin{bmatrix}
e_1(t) \\
e_2(t)
\end{bmatrix} = \frac{2}{\Delta t} \begin{bmatrix}
L_{11}(t) & L_{12}(t) \\
L_{12}(t) & L_{22}(t)
\end{bmatrix} \begin{bmatrix}
i_1(t) \\
i_2(t)
\end{bmatrix} + \begin{bmatrix}
e_{h1}(t) \\
e_{h2}(t)
\end{bmatrix}
\]

(4.23)

where the history terms are calculated as

\[
\begin{bmatrix}
e_{h1}(t) \\
e_{h2}(t)
\end{bmatrix} = -\frac{2}{\Delta t} \begin{bmatrix}
L_{11}(t-\Delta t) & L_{12}(t-\Delta t) \\
L_{12}(t-\Delta t) & L_{22}(t-\Delta t)
\end{bmatrix} \begin{bmatrix}
i_1(t-\Delta t) \\
i_2(t-\Delta t)
\end{bmatrix} - \begin{bmatrix}
e_1(t-\Delta t) \\
e_2(t-\Delta t)
\end{bmatrix}
\]

(4.24)

To shorten the notation, in the equations to follow \((t)\) is omitted and \((t - \Delta t)\) is replaced with \(\tilde{}\) to denote the values of variables from the preceding time step.

From the system of equations describing coupled stator (4.18) and rotor (4.19) windings of an induction machine, we can obtain the discretized equations for the
Chapter 4. Modelling with Multilevel MATE

branch voltages in a similar manner:

\[
\begin{bmatrix}
    e_a \\
    e_b \\
    e_c \\
    e_A \\
    e_B \\
    e_C
\end{bmatrix} =
\begin{bmatrix}
    R_a & 0 & 0 & 0 & 0 & 0 \\
    0 & R_b & 0 & 0 & 0 & 0 \\
    0 & 0 & R_c & 0 & 0 & 0 \\
    0 & 0 & 0 & R_A & 0 & 0 \\
    0 & 0 & 0 & 0 & R_B & 0 \\
    0 & 0 & 0 & 0 & 0 & R_C
\end{bmatrix}
\begin{bmatrix}
    i_a \\
    i_b \\
    i_c \\
    i_A \\
    i_B \\
    i_C
\end{bmatrix}
\]

\[
+ \frac{2}{\Delta t}
\begin{bmatrix}
    L_{aa} & L_{ab} & L_{ac} & L_{aA}(\theta_r) & L_{aB}(\theta_r) & L_{aC}(\theta_r) \\
    L_{ab} & L_{bb} & L_{bc} & L_{bA}(\theta_r) & L_{bB}(\theta_r) & L_{bC}(\theta_r) \\
    L_{ac} & L_{bc} & L_{cc} & L_{cA}(\theta_r) & L_{cB}(\theta_r) & L_{cC}(\theta_r) \\
    L_{aA}(\theta_r) & L_{bA}(\theta_r) & L_{cA}(\theta_r) & L_{AA} & L_{BA} & L_{CA} \\
    L_{aB}(\theta_r) & L_{bB}(\theta_r) & L_{cB}(\theta_r) & L_{AB} & L_{BB} & L_{BC} \\
    L_{aC}(\theta_r) & L_{bC}(\theta_r) & L_{cC}(\theta_r) & L_{AC} & L_{BC} & L_{CC}
\end{bmatrix}
\begin{bmatrix}
    i_a \\
    i_b \\
    i_c \\
    i_A \\
    i_B \\
    i_C
\end{bmatrix}
\]

Written in compact form we have:

\[
\begin{bmatrix}
    e_{abc} \\
    e_{ABC}
\end{bmatrix} = \left[ \begin{bmatrix}
    R_s & 0 \\
    0 & R_r
\end{bmatrix} + \frac{2}{\Delta t} \begin{bmatrix}
    L_s & L_{sr} \\
    L_{sr}^T & L_r
\end{bmatrix} \right] \begin{bmatrix}
    i_{abc} \\
    i_{ABC}
\end{bmatrix} + \begin{bmatrix}
    e_{ha} \\
    e_{hb} \\
    e_{hc} \\
    e_{hA} \\
    e_{hB} \\
    e_{hC}
\end{bmatrix}
\]

(4.25)

where the history terms are:

\[
\begin{bmatrix}
    e_{ha} \\
    e_{hb} \\
    e_{hc} \\
    e_{hA} \\
    e_{hB} \\
    e_{hC}
\end{bmatrix} =
\begin{bmatrix}
    R_a & 0 & 0 & 0 & 0 & 0 \\
    0 & R_b & 0 & 0 & 0 & 0 \\
    0 & 0 & R_c & 0 & 0 & 0 \\
    0 & 0 & 0 & R_A & 0 & 0 \\
    0 & 0 & 0 & 0 & R_B & 0 \\
    0 & 0 & 0 & 0 & 0 & R_C
\end{bmatrix}
\begin{bmatrix}
    i_0 \\
    i_0 \\
    i_0 \\
    i_0 \\
    i_0 \\
    i_0
\end{bmatrix}
\]

\[
\begin{bmatrix}
    e_{habc} \\
    e_{hABC}
\end{bmatrix} = \left[ \begin{bmatrix}
    R_s & 0 \\
    0 & R_r
\end{bmatrix} - \frac{2}{\Delta t} \begin{bmatrix}
    L_s & L_{sr} \\
    L_{sr}^T & L_r
\end{bmatrix} \right] \begin{bmatrix}
    i_{abc} \\
    i_{ABC}
\end{bmatrix} - \begin{bmatrix}
    e_{habc} \\
    e_{hABC}
\end{bmatrix}
\]

(4.26)
The above system of equations represents the branch voltage equations for coupled stator and rotor windings of an induction machine. This system of equations does not assume any particular connection of the stator and rotor windings (a grounded/ungrounded wye or a delta). The discretized phase-domain induction machine model can be visualized as depicted in Fig. 4.10.

\[ \text{Figure 4.10: Discrete phase domain induction machine electrical model} \]

### 4.4.1.2 Mechanical Equations of an Induction Machine

The system of equations (4.25) describing the electrical part of an induction machine contains terms \( L_{sr} \), which depend on the value of the rotor angular position \( \theta_r \) at a time instant \( t \). Since the rotor is rotating, the rotor angular position is constantly increasing and is related to the rotor angular velocity \( \omega_r \) as

\[ \omega_r = \frac{d\theta_r}{dt} \quad (4.27) \]

where \( \omega_r \) is obtained from the well-known relationship between the machine’s accelerating torque \( (T_a) \) and the rotor angular velocity \( (\omega_r) \)

\[ T_a = T_e - T_m = J \frac{d\omega_m}{dt} + D \omega_m = J \left( \frac{2}{p_f} \right) \frac{d\omega_r}{dt} + D \left( \frac{2}{p_f} \right) \omega_r \quad (4.28) \]

where

- \( T_e \) is the electromagnetic torque
Chapter 4. Modelling with Multilevel MATE

- $T_m$ is the mechanical torque
- $J$ is the polar moment of inertia of the rotor and the connected load
- $p_f$ is the number of poles of the machine
- $\omega_m$ is the angular velocity of the rotor in mechanical radians per second
- $D$ is the damping coefficient

The differential equations (4.27),(4.28), discretized with the implicit trapezoidal rule, become difference equations of the following form:

$$\theta_r(t) - \frac{\Delta t}{2} \omega_r(t) = \theta_r(t - \Delta t) + \frac{\Delta t}{2} \omega_r(t - \Delta t)$$

(4.29)

$$\left( \frac{4J}{p_f \Delta t} + \frac{2D}{p_f} \right) \omega_r(t) - T_e(t) + T_m(t) = \left( \frac{4J}{p_f \Delta t} - \frac{2D}{p_f} \right) \omega_r(t - \Delta t) + T_e(t - \Delta t) - T_m(t - \Delta t)$$

(4.30)

The electromagnetic torque $T_e$ is calculated from the energy stored in the machine coupling field $W_f$, which is in fact the energy stored in all mutual and self inductances of rotor and stator windings, excluding leakage inductances. The energy stored in the coupling field can be written in the following form:

$$W_f = \text{stator energy} + \text{mutual stator-rotor} + \text{rotor energy} =$$

$$\frac{1}{2} \cdot \begin{bmatrix} i_a & i_b & i_c \end{bmatrix} \cdot \begin{bmatrix} L_{ls} & 0 & 0 \\ 0 & L_{ls} & 0 \\ 0 & 0 & L_{ls} \end{bmatrix} \cdot \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} i_a & i_b & i_c \end{bmatrix} \cdot L_{sr}^{\text{(ref)}} \cdot \begin{bmatrix} i_a^{\text{(ref)}} \\ i_b^{\text{(ref)}} \\ i_c^{\text{(ref)}} \end{bmatrix} +$$

$$\frac{1}{2} \cdot \begin{bmatrix} i_A^{\text{(ref)}} & i_B^{\text{(ref)}} & i_C^{\text{(ref)}} \end{bmatrix} \cdot \begin{bmatrix} L_{lr}^{\text{(ref)}} & 0 & 0 \\ 0 & L_{lr}^{\text{(ref)}} & 0 \\ 0 & 0 & L_{lr}^{\text{(ref)}} \end{bmatrix} \cdot \begin{bmatrix} i_A^{\text{(ref)}} \\ i_B^{\text{(ref)}} \\ i_C^{\text{(ref)}} \end{bmatrix}$$

(4.31)

with rotor quantities referred to the stator side using the effective turns ratio of the stator and rotor windings, and indicated with the superscript $^{\text{(ref)}}$.

The electromagnetic torque $T_e$ is obtained by solving the following relationship

$$T_e = \left( \frac{p_f}{2} \right) \frac{\partial W_f}{\partial \theta_r}$$

(4.32)
which results in the following expression valid in the continuous and discrete time domains:

\[
T_e = \left(\frac{pf}{2}\right) \left[i_a \ i_b \ i_c\right] \cdot \frac{\partial}{\partial \theta_r} L^{(ref)}_{sr} \cdot \left[\begin{array}{c}
\dot{i}_A^{(ref)} \\
\dot{i}_B^{(ref)} \\
\dot{i}_C^{(ref)}
\end{array}\right] = \\
- \left(\frac{pf}{2}\right) \cdot L_{ms} \cdot \left[i_a \ i_b \ i_c\right] \cdot \left[
\begin{array}{ccc}
\sin \theta_r & \sin(\theta_r + 120^\circ) & \sin(\theta_r - 120^\circ) \\
\sin(\theta_r - 120^\circ) & \sin \theta_r & \sin(\theta_r + 120^\circ) \\
\sin(\theta_r + 120^\circ) & \sin(\theta_r - 120^\circ) & \sin \theta_r
\end{array}\right] \cdot \left[
\begin{array}{c}
\dot{i}_A^{(ref)} \\
\dot{i}_B^{(ref)} \\
\dot{i}_C^{(ref)}
\end{array}\right]
\]

(4.33)

4.4.1.3 Implementation of the Phase-Domain Induction Machine Model

A system of equations that completely describes the induction machine phase-domain model includes the electrical part, as described by branch equations (4.25), (4.26) and the mechanical part (4.29), (4.30), together with the nonlinear torque equation (4.33). These equations can be combined in a matrix representation of hybrid modified nodal equations of the induction machine in the following general form:

\[
\begin{bmatrix}
A & p_A & p_{A,\text{nonlin}} \\
p_A' & -z_A & 0 \\
p_{A,\text{nonlin}}' & 0 & -z_{A,\text{nonlin}}
\end{bmatrix}
\begin{bmatrix}
\dot{v}_A \\
\dot{i}_{A,\text{sublink}} \\
\dot{i}_{A,\text{nonlin}}
\end{bmatrix}
+ \begin{bmatrix}
p \\
0 \\
0
\end{bmatrix} \cdot \begin{bmatrix}
i_{A} \\
0 \\
0
\end{bmatrix} = \begin{bmatrix}
h_A \\
-V_{A,\text{sublink}} \\
-V_{A,\text{nonlin}}
\end{bmatrix}
\]

where

\[
h_{A,\text{ref}} = p_{A,\text{ref}} \cdot v_{\text{ref}}
\]

Using these equations, the circuit depicted in Fig. 4.11 is modelled using Multilevel MATE to test the phase-domain induction machine model. The circuit consists of a three-phase induction motor (subsystem A) connected to a three-phase sinusoidal voltage source (subsystem B) through links \(i_{as}, i_{bs}, i_{cs}\). Stator and rotor windings are delta- and wye-connected, respectively. For generality, the rotor side is modelled to allow access to the rotor terminals through links \(i_{Ar}, i_{Br}, i_{Cr}\). In this way, the rotor terminals can either be shorted or connected to an external network, as in the case of a doubly fed induction generator, without any modifications to the induction machine model.

The system of branch equations (4.25) describing the electrical part of the induction machine contains nonlinear terms which depend on the value of the rotor angular position \(\theta_r\) at time instant \(t\). With a sufficiently small integration step, the prediction
of the electrical angular velocity $\omega_r$ can be made using linear extrapolation from the past values

$$\omega_{r,\text{pred}}(t) = 2\omega_r(t - \Delta t) - \omega_r(t - 2\Delta t)$$

and used to calculate the value of the rotor angular position according to (4.29). This procedure is justified, as the mechanical system that includes the rotating masses has slower time responses (the order of seconds) than the electrical circuit, time responses of which are much faster (the order of milliseconds). The induction machine implementation with the Multilevel MATE concept uses the prediction of $\theta_r$, but the accuracy of this prediction can be checked and corrected by iterations if required.

With this in mind, the electrical and mechanical systems of equations are coupled only through the nonlinear torque equation relating the machine's electromagnetic torque $T_e$ to the rotor and stator phase currents. Therefore, subsystem $A$, representing the induction machine, is divided according to its nature into subsubsystems $A_1$ and $A_2$, where the former represents the electrical part, and the latter represents the mechanical part of the machine. The induction machine coupled branch equations are modelled as sublinks of subsystem $A$, and the nonlinear torque equation is modelled as a nonlinear sublink equation.

Following the process of building the system of nodal equations $A \cdot v_A = h_A$ we notice that the machine rotor nodes “$\text{A}$”, “$\text{B}$”, “$\text{C}$” and “$\text{N}$”, and stator nodes “$\text{a}$”, “$\text{b}$” and “$\text{c}$”, form ungrounded subsubsystems due to their connection to the links and/or sublinks. As a result, the conductance matrix of subsystem $A$ is zero. In such a case, one node of each ungrounded subsubsystem has to be declared as its reference.

Figure 4.11: Electrical circuit configuration for testing the phase-domain induction machine model implemented with Multilevel MATE

Chapter 4. Modelling with Multilevel MATE

54
point and solved at the links level. Since the ungrounded subsubsystems in this case consist of one node each, all nodes of the induction machine model are duplicated as reference points \( v_{\text{ref}} \) for their respective subsubsystems.

For the circuit in Fig. 4.11, nodal equations for the stator node "a" and the rotor node "A" (taking into account the delta connection of the stator windings and the wye connection of the rotor windings) can be written as

\[
\begin{align*}
&v_a + i_a - i_b - i_{as} - v_{\text{aref}} = 0 \\
&v_A + i_A - i_{Ar} - v_{\text{Aref}} = 0
\end{align*}
\]

Since \( v_a = v_{\text{aref}} \) and \( v_A = v_{\text{Aref}} \), these equations simply state that the sum of the link and sublink currents coming into nodes "a" and "A" is zero. By expanding this relationship to the remaining induction machine nodes, we obtain the following arrays and vectors associated with the nodal equations of the electrical part of the induction machine

\[
A_1 \cdot v_{A1} + p_{A1} \cdot i_{A1 \text{ sublink}} + p \cdot i_a = h_{A1} - h_{A, \text{ref}}
\]

expanded as

\[
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
v_a \\
v_b \\
v_c \\
v_A \\
v_B \\
v_C \\
v_N
\end{bmatrix}
+
\begin{bmatrix}
+1 & 0 & -1 & 0 & 0 & 0 \\
-1 & +1 & 0 & 0 & 0 & 0 \\
0 & -1 & +1 & 0 & 0 & 0 \\
0 & 0 & 0 & +1 & 0 & 0 \\
0 & 0 & 0 & 0 & +1 & 0 \\
0 & 0 & 0 & -1 & -1 & -1
\end{bmatrix}
\begin{bmatrix}
i_a \\
i_b \\
i_c \\
i_A \\
i_B \\
i_C
\end{bmatrix}
=
\begin{bmatrix}
0 \\
0 \\
0 \\
0 \\
0 \\
0
\end{bmatrix}
-
\begin{bmatrix}
-v_{\text{aref}} \\
v_{\text{bref}} \\
v_{\text{cref}} \\
v_{\text{Aref}} \\
v_{\text{Bref}} \\
v_{\text{Cref}} \\
v_{\text{Nref}}
\end{bmatrix}
\]

Pseudo nodal equations for the mechanical system can be written from (4.29), (4.30), to obtain the following arrays and vectors associated with the mechanical part of the induction machine:

\[
A_2 \cdot v_{A2} + p_{A2 \text{ nonlin}} \cdot i_{A2 \text{ nonlin}} = h_{A2}
\]
Chapter 4. Modelling with Multilevel MATE

and

\[
\begin{bmatrix}
1 & -\frac{\Delta t}{\omega_r} & 0 \\
0 & \frac{4\omega_r}{2p_f\Delta t} + \frac{2D}{p_f}\omega_r & -1
\end{bmatrix}
\begin{bmatrix}
\theta_r \\
\omega_r
\end{bmatrix}
+ \begin{bmatrix}
0 \\
-1
\end{bmatrix}
[T_e] = \begin{bmatrix}
-\frac{\theta_r + \frac{\Delta t}{2}\omega_r'}{\omega_r'} \\
\frac{4\omega_r}{2p_f\Delta t} - \frac{2D}{p_f}\omega_r + T_e' - T_m'
\end{bmatrix}
\]

The mechanical torque \( T_m(t) \) acts as a forcing function, and is therefore included on the right-hand side of the system of equations, which corresponds to sources.

The sublink equations of subsystem A represent the coupled branch equations of the machine written in the following form, which takes into account the stator and rotor winding connections:

\[ p_{A1}^i \cdot v_{A1} - z_{A1} \cdot i_{A1\text{-sublink}} = -V_{A1\text{-sublink}} \]  

In the expanded form:

\[
\begin{bmatrix}
+1 & 0 & -1 & 0 & 0 & 0 & 0 \\
-1 & +1 & 0 & 0 & 0 & 0 & 0 \\
0 & -1 & +1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & +1 & 0 & 0 & -1 \\
0 & 0 & 0 & 0 & +1 & -1 & 0 \\
0 & 0 & 0 & 0 & 0 & +1 & -1
\end{bmatrix}
\begin{bmatrix}
v_a \\
v_b \\
v_c \\
v_A \\
v_B \\
v_C \\
v_N
\end{bmatrix} = \begin{bmatrix}
\begin{bmatrix}
R_s & 0 \\
0 & R_r
\end{bmatrix} + \frac{2}{\Delta t} \begin{bmatrix}
L_s & L_{sr} \\
L_{sr}^T & L_r
\end{bmatrix}
\begin{bmatrix}
i_a \\
i_b \\
i_c \\
i_A \\
i_B \\
i_C
\end{bmatrix}
= -\frac{4}{\Delta t} \begin{bmatrix}
L_s & L_{sr} \\
L_{sr}^T & L_r
\end{bmatrix}
\begin{bmatrix}
i_a' \\
i_b' \\
i_c' \\
i_A' \\
i_B' \\
i_C'
\end{bmatrix} + \begin{bmatrix}
e_a' \\
e_b' \\
e_c' \\
e_A' \\
e_B' \\
e_C'
\end{bmatrix}
\]

Finally, the equation representing the nonlinear torque equation can be expressed in the following matrix form:

\[ -z_{A2\text{-nonlin}} \cdot i_{A2\text{-nonlin}} = -V_{A2\text{-nonlin}} \]  

and

\[
[1][T_e] = \left[-\frac{p_fL_{ms}}{2}\begin{bmatrix} i_a & i_b & i_c \end{bmatrix}\begin{bmatrix}
\sin\theta_r & \sin(\theta_r + 120^\circ) & \sin(\theta_r - 120^\circ) \\
\sin(\theta_r - 120^\circ) & \sin\theta_r & \sin(\theta_r + 120^\circ) \\
\sin(\theta_r + 120^\circ) & \sin(\theta_r - 120^\circ) & \sin\theta_r
\end{bmatrix}\frac{N_r}{N_s}\begin{bmatrix} i_A \\ i_B \\ i_C \end{bmatrix}\right]
\]

where \( N_r/N_s \) is the rotor winding to stator winding effective turns ratio.

To summarize, the induction machine phase-domain model represented as a six-
terminal device will have the following hybrid form:

\[
\begin{bmatrix}
    A_1 & 0 & p_{A1} & 0 & p_1 \\
    0 & A_2 & 0 & p_{A2,\text{nonlin}} & 0 \\
    p_{A1}^t & 0 & -z_{A1} & 0 & 0 \\
    0 & 0 & 0 & -z_{A2,\text{nonlin}} & 0
\end{bmatrix}
\begin{bmatrix}
    v_{A1} \\
    v_{A2} \\
    i_{A1,\text{sublink}} \\
    i_{A2,\text{nonlin}} \\
    i_\alpha
\end{bmatrix}
= 
\begin{bmatrix}
    0 \\
    h_{A2} \\
    -V_{A1,\text{sublink}} \\
    -V_{A2,\text{nonlin}} \\
    0
\end{bmatrix}
\begin{bmatrix}
    p_{A1,v_{\text{ref}}} \\
    0 \\
    0 \\
    0 \\
    0
\end{bmatrix}
\]

By applying the Multilevel MATE approach, we obtain the Multilevel MATE system of equations as follows:

\[
\begin{bmatrix}
    I & 0 & a_{A1} & 0 & a_1 \\
    0 & I & 0 & a_{A2,\text{nonlin}} & 0 \\
    0 & p_{A1}^t a_{A1} + z_{A1} & 0 & 0 & 0 \\
    0 & 0 & 0 & -z_{A2,\text{nonlin}} & 0
\end{bmatrix}
\begin{bmatrix}
    v_{A1} \\
    v_{A2} \\
    i_{A1,\text{sublink}} \\
    i_{A2,\text{nonlin}} \\
    i_\alpha
\end{bmatrix}
= 
\begin{bmatrix}
    0 \\
    e_{A2} \\
    +V_{A1,\text{sublink}} \\
    -V_{A2,\text{nonlin}} \\
    0
\end{bmatrix}
\begin{bmatrix}
    a_{A1,v_{\text{ref}}} \\
    0 \\
    0 \\
    0 \\
    0
\end{bmatrix}
\]

At every time step of the simulation, the induction machine model will perform the following procedure:

- predict the value of \( \theta_r \) using linear extrapolation (4.35)
- recalculate \([z_{A1}]\) based on the predicted value of \( \theta_r \):

\[
z_{A1} = \begin{bmatrix} R_s & 0 \\ 0 & R_r \end{bmatrix} + \frac{2}{\Delta t} \begin{bmatrix} L_s & L_{sr}(\theta_r) \\ L_{sr}^T(\theta_r) & L_r \end{bmatrix}
\]

- recalculate the contributions of the sublinks to the Thévenin equivalent of sub-subsystem \( A_1 \):

\[
\Delta a_1 = a_1 - a_{A1} \cdot (p_{A1}^t a_{A1} + z_{A1})^{-1} p_{A1}^t a_1 \\
\Delta e_{A1} = e_{A1} - a_{A1} \cdot (p_{A1}^t a_{A1} + z_{A1})^{-1} (p_{A1}^t e_{A1} + V_{A1,\text{sublink}})
\]

- calculate the modified Thévenin equivalent of subsystem \( A \) as seen from the linking nodes\(^6\) and submit it to the links \([p^a_{\text{MTE}}], [p^e_{\text{A,MTE}}]\), where

\[
a_{\text{MTE}} = a_1 - \Delta a_1 \\
e_{\text{A,MTE}} = e_{A1} - \Delta e_{A1}
\]

- receive the links currents \([i_\alpha]\) and reference voltages \([v_{\text{ref}}]\) from the links and

---

\(^6\)The Multilevel MATE algorithm will recognize that the mechanical system is decoupled from the electrical, therefore \( a_{\text{MTE}} = a_{\text{1,MTE}} \) and \( e_{\text{A,MTE}} = e_{\text{A1,MTE}} \)
recalculate its nodal voltages:

\[ v_A = e_{A,MTE} - a_{MTE}v_{ref} - a_{MTE}i_\alpha \]

\[ a_{MTE}v_{ref} = a_{A1vref} - \Delta a_{1vref} = a_{A1vref} - a_A (p^t_Aa_A + z_A) p^t_Aa_{A1vref} \]

- update the history terms \([h_{A2}]\) and \([V_{A1,sublink}]\) for the next simulation step

At every time step of the simulation the links system will perform the following procedure with respect to subsystem A:

- receive subsystem’s A modified Thévenin equivalent
- calculate \([z_{a,MTE}]\) and \([e_{a,MTE}]\) from
  \[ z_{a,MTE} = p^t_{a,MTE} + q^t b_{MTE} + z \]
  \[ e_{a,MTE} = p^t_{e,MTE} + q^t e_{B,MTE} + V_\alpha \]
- calculate the reference voltages of subsystem A from
  \[ v_{ref} = (p^t_{v_{ref}} \cdot z_{a,MTE}^{-1} \cdot p_{v_{ref}} - z_{v_{ref}})^{-1} \cdot (p^t_{v_{ref}} \cdot z_{a,MTE}^{-1} \cdot e_{a,MTE} - h_{v_{ref}}) \]
- calculate the link currents from
  \[ i_\alpha = z_{a,MTE}^{-1} \cdot (e_{a,MTE} - p_{v_{ref}} \cdot v_{ref}) \]
- return \([v_{ref}]\) and \([i_\alpha]\) to subsystem A.

The Multilevel MATE algorithm will automatically recognize that the mechanical system is decoupled from the electrical, therefore no iterations are required to solve the nonlinear torque equation. Furthermore, the induction machine model will appear as a 6x6 Thévenin equivalent as seen from the machine’s terminals (3x3 if the rotor terminals were shorted and not exposed to the network).

The phase-domain induction machine model presented here has been tested and successfully compared with results from the literature [30]. The machine parameters are given in Appendix D.1. In the following graphs, results are presented first for the free acceleration of a 2250 hp induction motor from stall, and then for step changes in load torque from zero to nominal (8900 Nm) at 0.5 s and from nominal back to zero at 2.5 s, with the machine initially operating at synchronous speed.
Chapter 4. Modelling with Multilevel MATE

Figure 4.12: Stator phase current during free acceleration of a 2250 hp induction motor

Figure 4.13: Rotor phase current during free acceleration of a 2250 hp induction motor
Chapter 4. Modelling with Multilevel MATE

Figure 4.14: Electromagnetic torque during startup of a 2250 hp induction motor

Figure 4.15: Rotor speed during startup of a 2250 hp induction motor
Chapter 4. Modelling with Multilevel MATE

Figure 4.16: Torque-speed characteristics during startup of a 2250 hp induction motor

Figure 4.17: Torque-speed characteristics during step changes in load torque of a 2250 hp induction motor
Figure 4.18: Stator phase current during step changes in load torque of a 2250 hp induction motor

Figure 4.19: Rotor phase current during step changes in load torque of a 2250 hp induction motor
Figure 4.20: Electromagnetic torque during step changes in load torque of a 2250 hp induction motor

Figure 4.21: Rotor speed during step changes in load torque of a 2250 hp induction motor

63
4.4.2 Phase-Domain Synchronous Machine Model

A phase-domain model of a synchronous machine [25] has been implemented in OVNI in an analogous manner to the induction machine model, using sublink equations and the Multilevel MATE concept. The synchronous machine model is composed of the stator phase windings \((a, b, c)\), equivalent to those of an induction machine, and the field rotor winding \((fd)\) supplied by a DC excitation voltage. In addition, the synchronous machine model has equivalent damper windings in the \(d\) and \(q\) axes to model the paths for induced rotor currents. The number of damper windings varies depending on the type of synchronous machine to be represented. A generic diagram of the electrical part of a wye-connected synchronous machine is depicted in Fig. 4.22. The modelled machine is a steam turbine synchronous generator with non-salient poles. The equations representing the electrical part are derived in a similar manner as described for the induction machine model in the preceding section. The mechanical equations of the synchronous machine are equivalent to those of the induction machine.

![Figure 4.22: Rotor and stator circuits of a synchronous machine](image)

The phase-domain synchronous machine model implemented with Multilevel MATE has been tested and successfully compared against results from the literature [30]. The machine parameters are given in Appendix D.2. Results of dynamic performance of a 835 MVA steam turbine generator during a step increase in input torque from zero to fifty percent of the rated value are shown in Fig. 4.23 to Fig. 4.28.
Figure 4.23: Stator phase current during a step increase in input torque of a 835 MVA steam turbine generator

Figure 4.24: Field current during a step increase in input torque of a 835 MVA steam turbine generator
Figure 4.25: Electromagnetic torque during a step increase in input torque of a 835 MVA steam turbine generator

Figure 4.26: Rotor speed during a step increase in input torque of a 835 MVA steam turbine generator
Figure 4.27: Rotor angle during a step increase in input torque of a 835 MVA steam turbine generator

Figure 4.28: Torque-rotor angle characteristics during a step increase in input torque of a 835 MVA steam turbine generator
4.4.3 Commentary on Modelling of Electric Machines

Our mathematical description of the implementation of a phase-domain machine model with Multilevel MATE may give rise to questions about how the efficiency of computation for these models can be evaluated. The main disadvantage of a phase-domain machine model is that its impedance matrix is time-dependent due to rotation of the rotor. Furthermore, if nodal analysis is used, time-dependent coupled branch equations have to be transformed into nodal equations at every time step. Even if modified nodal analysis is used, the machine's branch equations will increase the size of the system of equations to be solved, and cause the matrix of coefficients to change at every time step of the simulation. In the approach proposed in this thesis, the changing impedance matrix of the machine modelled with branch equations is recalculated at every time step separately from the subsystem's nodal equations, therefore the subsystem's nodal admittance matrix does not need to be inverted at every time step. The contribution of the machine's branch equations to the subsystem's nodal equations is integrated at the level of the subsystem's Thévenin equivalent (see equations (4.43), (4.44)) which makes this approach more efficient than the MNA or nodal analysis implementation approaches.

Implementation of the phase-domain machine models in the EMTP-type of simulators provides significant advancement over the commonly used \(dq0\) model with respect to the simulation time step necessary to interface the machine model and the network equations. In this work the simulation step of 1 ms was used, whereas a typical time step used in the EMTP simulations with a \(dq0\) model is in the order of 50 \(\mu s\) or less. Small time step with the \(dq0\) model is necessary to avoid instabilities due to predictions used in interfacing the model with a three-phase network representation.

4.5 Modelling of Controllers

A traditional approach to simulating controllers in power system networks uses an artificial time step delay between the network solution and the solution of the control systems to decouple the two systems. In recent years, different approaches have been proposed to achieve a simultaneous solution of the network and controller equations [31],[32],[33],[34]. Within OVNI, controller equations of the form

\[
A_{\text{ctrl}} \cdot x = h_{\text{ctrl}}
\]

(4.45)

where

- \([A_{\text{ctrl}}]\) is a matrix of controller coefficients
- \([x]\) is a vector of controller variables (inputs + outputs + internal variables)
• \([h_{\text{ctrl}}]\) is a vector of controller history terms

can be solved simultaneously using a hybrid system of nodal and branch equations describing the electric network.

Controllers are generally represented by block diagrams showing interconnections among different control blocks (elements), such as transfer functions, limiters, etc. Discretized equations of controller elements can be obtained in OVNI by mapping the Laplace operator \(s\) to the \(z\)-domain using the bilinear transformation (trapezoidal discretization), with a discrete time step \(\Delta t\):

\[
s = \frac{2}{\Delta t} \frac{1 - z^{-1}}{1 + z^{-1}}
\]  

(4.46)

For example, discretization of the first-order transfer function of the general form

\[
H(s) = K \frac{b_0 + b_1 s}{a_0 + a_1 s} = \frac{X_2(s)}{X_1(s)}
\]  

(4.47)

where \(X_1(s)\) and \(X_2(s)\) are the input and output of a transfer function in the Laplace domain, is obtained by substituting (4.46) into (4.47). The following first-order difference equation is obtained

\[-B_0 x_1(t) + A_0 x_2(t) = h(t)\]  

(4.48)

with

\[h(t) = B_1 x_1(t - \Delta t) - A_1 x_2(t - \Delta t)\]  

(4.49)

and

\[
A_0 = a_0 + a_1 \frac{2}{\Delta t} \\
A_1 = a_0 - a_1 \frac{2}{\Delta t} \\
B_0 = K \left( b_0 + b_1 \frac{2}{\Delta t} \right) \\
B_1 = K \left( b_0 - b_1 \frac{2}{\Delta t} \right)
\]

The term \(h(t)\) in (4.49) depends on the values of the input and output from the previous time step, therefore it is referred to as a history term of a first-order transfer function.

Controller equations at the subsystem level (local control) can be incorporated into the MATE system of equations as functional sublinks. In the case of nonlinear controllers containing, for example, limiters, their equations can be treated as nonlinear sublinks as described in Section 3.6. In the most general representation, the
controller equations at the subsystem level will have the following form:

\[
\begin{bmatrix}
A & p_A & p_{ctrl} \\
p_A^t & -z_A & 0 \\
p_{ctrl}^t & -z_{A,ctrl} & -z_{ctrl}
\end{bmatrix}
\begin{bmatrix}
0 \\
p \\
p^t
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_A \\
v_B \\
\dot{v}_B \\
\dot{z}
\end{bmatrix}
= 
\begin{bmatrix}
h_A \\
h_A \\
h_B \\
h_B
\end{bmatrix}
\]

(4.50)

where

- \([p_{ctrl}]\) is a connectivity array of controllers in subsystem A,

and

\[
p_{ctrl}^t \cdot v_A - z_{A,ctrl} \cdot i_A - z_{ctrl} \cdot i_{ctrl} = -V_{ctrl}
\]

(4.51)

represents the system of controller equations where \([p_{ctrl}^t]\), generally speaking, is not the transpose of the connectivity array \([p_{ctrl}]\). As can be noticed from (4.50), the controller equations introduce asymmetry into the sublink equations.

If a controller operates at the system level (global control), its equations can be incorporated into the MATE system of equations as links. Such a controller uses global measurements of power system variables from multiple subsystems as inputs and/or controls variables from multiple subsystems as outputs. A global controller would provide, for example, damping of inter-area oscillations.

The third type of controller is implemented at the level of a power system component (component control). Its action is not transparent to the subsystem, but it is embedded in the information passed by the power system component to the subsystem.

The implementation of controllers in OVNI using Multilevel MATE is demonstrated in the following example and described in [33].

4.5.1 Example: Modelling of a Single-Machine Infinite-Bus System in \(dq\) Coordinates with Multilevel MATE

A single-machine infinite-bus power system from [27] is modelled using the Multilevel MATE approach. The system consists of an equivalent synchronous generator representing four generating units of a power plant, a step-up transformer and a double-circuit transmission line connecting the generator to a large power system represented by an infinite bus. The synchronous generator is modelled in the \(dq\) reference frame, including the rotor dynamics. The model includes a bus-fed thyristor
excitation system with an automatic voltage regulator (AVR) and power system stabilizer (PSS). In the example, network transients and stator transients are neglected as is traditionally done in power system transient stability analyses. The electrical discrete-time circuit of the example system is shown in Fig. 4.29.

Figure 4.29: Single-machine infinite-bus equivalent network

The bus-fed thyristor excitation system with the AVR and the PSS controller are shown in Fig. 4.30. Nonlinear equations associated with the limiters are described as:

\[
\begin{align*}
-x_3(t) + x_S(t) &= 0 \quad \text{for} \quad x_{S_{\text{min}}} \leq x_S \leq x_{S_{\text{max}}} \\
-x_3(t) &= x_{S_{\text{max}}} \quad \text{for} \quad x_{S_{\text{max}}} < x_S \\
-x_3(t) &= x_{S_{\text{min}}} \quad \text{for} \quad x_S < x_{S_{\text{min}}}
\end{align*}
\]
\[
\begin{align*}
\begin{cases}
E_{fd}(t) + K_A(x_1(t) - x_S(t)) = K_A V_{ref} & \text{for } E_{F_{\min}} \leq E_{fd} \leq E_{F_{\max}} \\
E_{fd} = E_{F_{\max}} & \text{for } E_{F_{\max}} < E_{fd} \\
E_{fd} = E_{F_{\min}} & \text{for } E_{fd} < E_{F_{\min}}
\end{cases}
\end{align*}
\]

A complete system of equations of the excitation system discretized using the trapezoidal rule (bilinear transformation) is shown in (4.52).

\[
\begin{bmatrix}
E_t(t) \\
\Delta \omega_r(t) \\
E_{fd}(t) \\
x_1(t) \\
x_2(t) \\
x_3(t) \\
x_S(t)
\end{bmatrix} = \begin{bmatrix}
K_A (0,0) & 0 & 0 & -K_A (0,0) \\
-1 & 0 & 0 & (1 + T_{R_{\Delta t}}) & 0 & 0 & 0 \\
0 & -K_{STAB} (TW \frac{2}{\Delta t}) & 0 & 0 & (1 + TW \frac{2}{\Delta t}) & 0 & 0 \\
0 & 0 & 0 & 0 & - (1 + T_1 \frac{2}{\Delta t}) & (1 + T_2 \frac{2}{\Delta t}) & 0 \\
0 & 0 & 0 & 0 & 0 & -1 (0,0) & 1
\end{bmatrix}
\]

where

- \( E_t \) is the synchronous machine stator terminal voltage
- \( \Delta \omega_r \) is the per unit relative rotor angular velocity
- \( E_{fd} = \frac{L_{\text{ad}}}{R_{fd}} E_{fd} \) is the exciter output voltage

Multilevel MATE system partitioning is next applied to separate the main components of the system: the network and the synchronous machine with the exciter.

The system partitioning structure is shown in Fig. 4.31, where

- \([v_A]\) represents the network’s nodal voltages in the dq reference frame
- \([i_A]\) represents the network’s branch currents for ideal switches \((i_{sw1}, i_{sw2} \cdots \) in Fig. 4.29) in the dq reference frame
- \([v_B]\) represents the synchronous machine stator and rotor voltages, linear mechanical variables \((\delta, \Delta \omega_r)\) and linear exciter variables \((x_1, x_2 \text{ and } x_3 \text{ in Fig. 4.30})\)
Chapter 4. Modelling with Multilevel MATE

Figure 4.31: Multilevel MATE partitioning of the single machine infinite bus system

- \([i_B]\) represents the synchronous machine stator and rotor currents, nonlinear mechanical variables \(T_e\) and nonlinear limited exciter variables \(x_s, E_{fd}\) and \(E_t\) in Fig. 4.30

- \([i_\alpha]\) represents the terminal \(dq\) link currents and the rotor angle \(\delta\)

The synchronous machine model in the \(dq\) reference frame is better described by branch equations, as with the phase-domain model. The machine's branch equations are included as functional sublinks of subsystem \(B\) representing the synchronous machine. The stator-side nodal equations are derived in a similar way as for the phase-domain induction machine model (4.36). The rotor-side nodal equations state that the nodal voltage of the field winding is equal to the exciter voltage (with appropriate scaling) and that the nodal voltages of the damper windings are zero. The nonlinear equation for calculating the terminal voltage \(E_t\):

\[
E_t = \sqrt{e_d^2 + e_q^2}
\]

as well as the nonlinear torque equation and the equations of the exciter's limiters are included as nonlinear sublinks of subsystem \(B\) and are solved by iterations at every time step.

Subsystem \(A\) represents the connecting network with the infinite bus, all variables decomposed into \(dq\) coordinates. Since the subsystem does not have a direct connection to ground (only through the sublinks and the ideal voltage source representing the infinite bus), the procedure described earlier for ungrounded subsystems is applied to reference one nodal voltage for each ungrounded subs subsystem. The
nonlinear equations associated with the network relate the infinite bus voltage to the rotor angle \( \delta \) as

\[
\begin{align*}
\epsilon_{Bd} &= E_B \sin \delta \\
\epsilon_{Bq} &= E_B \cos \delta
\end{align*}
\]

The transient response of the system variables to a three-phase short circuit applied on Circuit 2 of the transmission line is simulated using the Multilevel MATE approach. The results for the system's currents and voltages, as well as for the synchronous machine's mechanical variables, are depicted in Fig. 4.32, 4.33. The results agree well with the time responses presented in [27].

### 4.5.2 Commentary on the Simultaneous Solution of Controller Equations

A time delay between the solution of control and network equations can cause instability, inaccuracy and numerical oscillations [31]. These problems are well documented in the literature, with different solutions suggested to either reduce or overcome them. In the case of the EMTP-based programs, implementation of the TACS [35] module is an example of a non-simultaneous approach to solving control equations.

The approach to the simultaneous solution of control and network equations described in this thesis takes advantage of the proposed Multilevel MATE concept and modelling with functional sublinks. As nonlinearities iterate individually only against the system of links, which has to be optimally small due to requirements for the multimachine implementation of OVNI [10], the framework for iterating nonlinear equations is computationally efficient.

For linear controller equations, no iterations are necessary. For nonlinear controller equations, iterations will be performed within the framework described in Section 3.6. With the fixed point type of iteration, an average value of one to two iterations were needed in test cases described in this thesis, with the number increasing to about five to ten iterations for a few time steps following a sudden dramatic change of system conditions (for example, a short circuit, or during initialization from zero initial conditions, Section 3.6.1). The results published in [34] using Newton iteration show iteration counts of the same order as those obtained in this work. Newton iterations, however, have a higher solution overhead due to re-calculation of a Jacobian matrix at each simulation time step.
Figure 4.32: Transient response of the single-machine infinite-bus system with manual control: (a) terminal voltage, (b) active power, (c) rotor angle.
Figure 4.33: Transient response of the single-machine infinite-bus system with AVR/PSS: (a) terminal voltage, (b) active power, (c) rotor angle
Chapter 4. Modelling with Multilevel MATE

4.6 Summary

The concept of Multilevel MATE is further explored in this chapter. New solutions are presented for modelling of ideal voltage sources and ungrounded circuits and subsystems. The modelling of electric machines with branch equations by means of sublinks is then described. Finally, modelling of controllers with Multilevel MATE is explained and demonstrated through an example.
Chapter 5

Modelling and Simulation of a DFIG Wind Turbine System in OVNI

5.1 Introduction

Doubly fed induction generator (DFIG) wind turbines are increasingly used in new wind turbine installations all over the world. Growing concerns about the impact of a large number of wind turbine generators (WTG) on transient and voltage stability of power system networks has led engineers to revisit modelling and simulation practices traditionally used for system stability analyses. Because OVNI is based on the EMTP methodology for accurate detailed modelling, and the Multilevel MATE concept, which, combined with hardware solutions, allows for fast simulation of large power system networks, it represents an ideal tool for testing and developing benchmark models of different wind turbine installations.

The advantages of the detailed modelling and simulation of DFIG wind turbine systems with OVNI include: (1) three-phase representation allowing for accurate simulation of balanced and unbalanced network conditions; (2) phase-domain full-detail machine modelling, including the modelling of torsional shaft oscillations; (3) modelling of the WTG control systems; (4) and the modelling of power electronic voltage source inverters. The purpose of this chapter is to demonstrate how an experimental DFIG wind turbine system from [36] is modelled in OVNI and to describe solutions for modelling challenges associated with the full time-domain (EMTP-type) simulation of power systems. Finally, the results of the simulations are compared against traditional stability analysis simulation results obtained with the Transient Stability Assessment Tool (TSAT).

The research contributions reported in this chapter include the following:

- *Description and implementation of detailed modelling of a doubly fed induction generator wind turbine system with Multilevel MATE.*

- *Comparison of the OVNI simulation results against the experimental results and transient stability simulation.*
5.2 Wind Turbine Generators

Variable-speed turbines with doubly fed induction generators have become a new standard for wind turbines of installed capacity above 2 MW. As the ratings of such wind farms connected to the power system grid become closer to the ratings of traditional generating units, and their share in the total installed generating capacity of the power system becomes considerable, it becomes necessary to perform studies of the impact of large wind farm connections to the power system network. The main concern is to study the responses of wind farms to power system faults and their impact on overall system stability.

Variable-speed operation of the turbines is achieved through the use of power electronics converters that can also be used to improve the grid integration aspects. It is anticipated that in the future it may be possible to request specific responses of wind farms to network disturbances from the manufacturers to help system recovery. Detailed model parameters of converter-controlled wind turbines can only be provided by manufacturers, and these control details are usually confidential and not readily available. However, efforts are being made to create reasonably accurate general models of doubly fed induction generators that can produce realistic results of wind farm responses to system disturbances and the influence of the associated controls.

Fixed-speed wind turbines are easier to model, as their responses mainly depend on the nature of their components, such as the response of induction generators, shaft stiffness and inertia of the rotor parameters, the details of which are normally available.

There are four types of wind turbines in use today:

- **Fixed-speed wind turbine** with a conventional induction generator connected directly to the grid. This turbine was originally only stall-regulated, but in recent years the larger MW-size turbines have had an active pitch\(^7\) mechanism as well.

- **Pitch-regulated variable rotor resistance induction generator** has a variable rotor resistance that can vary the slip in the range of 2-10\% and allow a sufficiently variable speed operation to reduce output power fluctuations.

- **Doubly fed induction generator** has become a new standard wind turbine for larger units. A slip-ringed induction machine is used as a generator, with a converter connected between the rotor slip rings and the network. The stator of the machine is directly connected to the network.

---

\(^{7}\)Wind turbine blades have a controllable pitch angle to turn the blades out of the wind or into the wind.
5.3 Doubly Fed Induction Generator Wind Turbines

Physically, the machine of a doubly fed induction generator is a conventional wound rotor induction machine. The key distinction is that this machine is equipped with a solid-state AC excitation system. The AC excitation is supplied through an AC-DC-AC voltage converter. Doubly fed induction machines have a significantly different dynamic behaviour than conventional synchronous or induction machines.

The fundamental frequency electrical dynamic performance of a doubly fed induction generator is completely dominated by the field converter. Conventional aspects of the generator's performance related to the rotor angle, excitation voltage and synchronism are largely irrelevant. The electrical behaviour of the generator and converter is that of a current-regulated voltage-source inverter. The converter makes the wind-turbine behave like a voltage behind a reactance that produces the desired active and reactive current delivered to the device terminals. A schematic of a doubly fed wind turbine system with two voltage-source inverters and accompanying controls is depicted in Fig. 5.1.

5.4 DFIG Model Structure

To construct a DFIG wind turbine system model, the following model structure is normally considered:

- Doubly fed induction generator model based on a phase-domain induction machine model
- Voltage converter model based on an average model of a back-to-back PWM voltage-source inverter with stator and rotor-side converter control
- Wind model that maps the wind speed to the shaft mechanical power for the turbine. Emulation of wind disturbances such as gusts and ramps by varying input wind speed to the wind-power module
- Crowbar protection
5.5 Doubly Fed Induction Generator Model

A doubly fed induction generator can be modelled as an induction machine in phase coordinates. The difference between a conventional induction machine and a doubly fed induction machine is that the rotor of a doubly fed induction machine is connected to the grid via a voltage converter. The electrical equations of the two machines are identical, as discussed in Section 4.4.1. With respect to the mechanical equations, the detailed modelling of DFIG wind turbines requires a two-mass shaft representation, which we describe in the following subsection.

5.5.1 Two-mass Representation of a DFIG WTG Shaft

Torsional shaft oscillations of variable-speed wind generators are not transferred to the grid under normal operating conditions due to the fast active power control of the converters. In this case, a single-mass shaft model representation is usually sufficient. However, when the system response to heavy disturbances is analyzed (such as short circuits in the network), the generator and turbine acceleration can be simulated with sufficient accuracy only if shaft oscillations are included in the model. In this case, the
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

shaft has to be approximated by at least a two-mass model, one mass representing the
turbine inertia and the other mass representing the generator inertia. The inertia of
the gear-box is usually not modelled separately but rather included in the generator
inertia.

The mechanical coupling of the turbine and generator through the gear-box can
be represented by two spring-connected rotating masses described by the rotational
form of Newton’s second law [28]:

\[
[J] \frac{d^2 \theta}{dt^2} + [D] \frac{d \theta}{dt} + [K] \theta = [T_t] - [T_g]
\]

where

- \([J]\) is a diagonal matrix of the turbine and generator moments of inertia \((J_t, J_g)\)
- \([\theta]\) is a vector of angular positions
- \([\omega]\) is a vector of angular velocities
- \([D]\) is a matrix of damping coefficients
- \([K]\) is a matrix of stiffness coefficients (spring constants)
- \([T_t]\) is a vector representing the turbine mechanical torque
- \([T_g]\) is a vector representing the generator electromagnetic torque

![Figure 5.2: Two-mass representation of a DFIG WTG shaft for accurate simulation of torsional shaft oscillations](image)

Figure 5.2: Two-mass representation of a DFIG WTG shaft for accurate simulation
of torsional shaft oscillations
The above equation for the two mass turbine-generator coupling depicted in Fig. 5.2 can be written as:

\[
\begin{bmatrix}
J_t & 0 \\
0 & J_g
\end{bmatrix}
\frac{d^2}{dt^2}
\begin{bmatrix}
\theta_t \\
\theta_g
\end{bmatrix}
+
\begin{bmatrix}
D_t + D_{tg} & -D_{tg} \\
-D_{tg} & D_g + D_{tg}
\end{bmatrix}
\frac{d}{dt}
\begin{bmatrix}
\theta_t \\
\theta_g
\end{bmatrix}
+
\begin{bmatrix}
K_{tg} & -K_{tg} \\
-K_{tg} & K_{tg}
\end{bmatrix}
\begin{bmatrix}
\theta_t \\
\theta_g
\end{bmatrix}
=
\begin{bmatrix}
T_t \\
0
\end{bmatrix}
- \begin{bmatrix}
0 \\
T_g
\end{bmatrix}
\tag{5.2}
\]

which when substituted with \([\omega] = [\omega_t \omega_g] \) (5.2) becomes

\[
T_t = J_t \frac{d\omega_t}{dt} + D_t \omega_t + D_{tg}(\omega_t - \omega_g) + K_{tg}(\theta_t - \theta_g)
\]

\[
-T_g = J_g \frac{d\omega_g}{dt} + D_g \omega_g - D_{tg}(\omega_t - \omega_g) - K_{tg}(\theta_t - \theta_g)
\tag{5.3}
\]

These two equations can be written in the following form:

\[
T_m - T_g = J_g \frac{d\omega_g}{dt} + D_g \omega_g
\tag{5.4a}
\]

\[
T_t - T_m = J_t \frac{d\omega_t}{dt} + D_t \omega_t
\tag{5.4b}
\]

where

\[
T_m = D_{tg}(\omega_t - \omega_g) + K_{tg}(\theta_t - \theta_g)
\tag{5.5}
\]

Equation (5.4a) represents the generator inertia and is equivalent to (4.28). The shaft model represents the turbine inertia (5.4b) and the coupling between the turbine and the generator (5.5). The difference equations describing the generator and turbine's inertia are produced analogously to (4.30) presented in Section 4.4.1.2. Note that the turbine torque \(T_t\) is calculated from the mechanical power extracted from the wind using the following equation:

\[
T_t = \frac{P_{\text{wind}}}{\omega_t}
\tag{5.6}
\]

### 5.6 Voltage Converter Model and Control

The converter and its controls highly influence the dynamic response of a DFIG. In this section, rotor- and stator-side converter modelling will be described, as well as the modelling of their controls. The models represent typical equipment and control structures derived from [36].

The voltage converter that supplies the rotor of a doubly fed induction generator consists of two voltage-source inverters linked via a DC-link capacitor, as shown in Fig. 5.1. The voltage-source inverter is connected to the network and to the DFIG stator terminals. It is, therefore, referred to as a stator-side PWM converter. The
voltage-source inverter connected to the DFIG rotor circuit is referred to as a rotor-side PWM converter. The rotor and stator-side PWM converters are self commutated converters and are set up by six pulse bridges, as depicted in Fig. 5.3.

Assuming an ideal DC voltage and an ideal PWM modulation (infinite modulation frequency), the fundamental frequency line-to-line stator-side converter voltage RMS value ($V_{s1\_line}$) and the DC voltage ($v_{DC}$) can be related to each other as

$$V_{s1\_line} = \frac{\sqrt{3}}{2\sqrt{2}} P_m v_{DC}$$  \hspace{1cm} (5.7)

with a similar expression for the rotor-side converter voltage-transfer characteristic. The pulse-width modulation factor $P_m$ is the control variable of the stator-side PWM converter. The converter model is completed by the power-conservation equation (assuming a lossless converter) expressed for the stator side as

$$v_{DC}i_{DC} - \sqrt{3} \text{Re}(\bar{V}_{s1\_line}\bar{i}_{s1\_line}^*) = 0$$  \hspace{1cm} (5.8)

The rotor side has a similar expression, where $\bar{V}_{s1\_line}$ and $\bar{i}_{s1\_line}$ are phasor quantities of stator-side converter AC line voltage and current, and * denotes a complex-conjugate value.

The switching frequency of PWM converters is usually several hundreds Hz, and the average switching losses are proportional to the square of $v_{DC}$. The switching losses can be taken into account by including a resistance between the two DC terminals shown in Fig. 5.3.

The approximate fundamental frequency modelling approach of PWM converters (average model) neglects all switching operations occurring within the voltage-source inverters and represents the converter as an ideal, lossless DC-to-fundamental-frequency AC converter complying with expression (5.7). According to the results presented in several publications, the high-frequency ripple due to switching harmonics caused by the PWM operation of the voltage converter is of no significance for
studying the performance of DFIG WTG in response to network events [37], and in practice is small and further limited by the inclusion of supply-side inductors [36]. Hence, in this example the average model of PWM converters is considered sufficient. Because the induction generator and the network are both modelled in phase coordinates (\(abc\) reference frame), the converter equations are modelled in phase coordinates as well.

### 5.6.1 Stator-side Converter Model

When deriving the equations of the stator-side converter, it is assumed that the converter is connected to the grid via a three-phase line with per phase inductance \(L\) and resistance \(R\) (see Fig. 5.1). The voltage balance across the line is

\[
\begin{bmatrix}
  v_a \\
v_b \\
v_c
\end{bmatrix} = R \begin{bmatrix}
  i_{a1} \\
i_{b1} \\
i_{c1}
\end{bmatrix} + L \frac{d}{dt} \begin{bmatrix}
  i_{a1} \\
i_{b1} \\
i_{c1}
\end{bmatrix} + \begin{bmatrix}
  v_{a1} \\
v_{b1} \\
v_{c1}
\end{bmatrix}
\]  

(5.9)

where \(v_a, v_b, v_c\) are the DFIG stator terminal phase voltages, \(v_{a1}, v_{b1}, v_{c1}\) are the stator-side converter AC terminal phase voltages, and \(i_{a1}, i_{b1}, i_{c1}\) are the stator-side converter AC terminal phase currents.

The power conservation equation between the DC link and the stator-side converter, with losses neglected, can be written in phase coordinates as

\[
P_{\text{conv}} = v_{a1}i_{a1} + v_{b1}i_{b1} + v_{c1}i_{c1} = v_{\text{DC}}i_{\text{DC}}
\]

(5.10)

where all quantities are instantaneous values and \(P_{\text{conv}}\) is the three-phase active power of the voltage converter. The calculation of active power from instantaneous values of phase currents and voltages is valid under all operating conditions, transient and steady state operation, balanced and unbalanced conditions and sinusoidal or non-sinusoidal waveforms [38].

The stator-side converter is vector controlled in the stator voltage-oriented \(dq\) reference frame. The angular position of the stator voltage vector \(V_s\) can be calculated from the stationary \(\alpha\beta\) stator voltage components. Transformation from the stationary \(abc\) reference frame to the stationary \(\alpha\beta\) reference frame (5.11) is done using Clarke’s transformation (Appendix B.1). The vectors of corresponding stator voltages are shown in Fig. 5.4. The variables in the reference frame are scaled to have the same magnitude as the original phase-domain variables. A scaling factor of 2/3 is used in the equations.

\[
\begin{bmatrix}
v_{s\alpha} \\
v_{s\beta}
\end{bmatrix} = \begin{bmatrix}
  2 \cos \left( \frac{2\pi}{3} \right) & \cos \left( \frac{4\pi}{3} \right) \\
  3 \sin \left( \frac{2\pi}{3} \right) & \sin \left( \frac{4\pi}{3} \right)
\end{bmatrix} \begin{bmatrix}
v_a \\
v_b \\
v_c
\end{bmatrix}
\]

(5.11)
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

The angular position $\theta_s$ of the stator voltage vector $V_s$ can be calculated in the most general case as

$$\theta_s = \int \omega_s \, dt = \tan^{-1} \frac{v_{\beta}}{v_{\alpha}}$$

(5.12)

where $\omega_s$ is the angular frequency of the stator voltage. This equation can be written in terms of the stator phase voltages using the inverse Clarke transformation as

$$\theta_s = \tan^{-1} \left( \frac{\sqrt{3}v_b - \sqrt{3}v_c}{v_a - \frac{1}{2}v_b - \frac{1}{2}v_c} \right)$$

(5.13)

In the particular situation considered here, the stator voltages' magnitude and frequency are dictated by the infinite bus. Therefore, the stator voltage's angular frequency is constant, and $\theta_s = \omega_s t$.

Transformation of the phase variables in the stationary $abc$ reference frame to a
rotating stator voltage oriented $dq$ reference frame, where $dq$ variables are scaled to have the same amplitude as the RMS value of phase variables, can be written for the stator voltages as

$$
\begin{bmatrix}
    v_d \\
    v_q
\end{bmatrix} = \frac{1}{\sqrt{2}} \cdot \frac{2}{3} \begin{bmatrix}
    \cos \theta_s & \cos \left(\theta_s - \frac{2\pi}{3}\right) & \cos \left(\theta_s + \frac{2\pi}{3}\right) \\
    -\sin \theta_s & -\sin \left(\theta_s - \frac{2\pi}{3}\right) & -\sin \left(\theta_s + \frac{2\pi}{3}\right)
\end{bmatrix} \begin{bmatrix}
    v_a \\
    v_b \\
    v_c
\end{bmatrix}
$$

The $d$-axis of the rotating $dq$ reference frame is aligned along the position of the stator voltage vector $V_s$, as shown in Fig. 5.4. The equations relating stator voltages in the $abc$ phase-domain frame and the $dq$ rotating reference frame can be written using the inverse $dq$ transformation as

\begin{align*}
    v_a &= \sqrt{2} V_s \cos (\omega_s t) = \sqrt{2} v_d \cos \theta_s - \sqrt{2} v_q \sin \theta_s \\
    v_b &= \sqrt{2} V_s \cos \left(\omega_s t - \frac{2\pi}{3}\right) = \sqrt{2} v_d \cos \left(\theta_s - \frac{2\pi}{3}\right) - \sqrt{2} v_q \sin \left(\theta_s - \frac{2\pi}{3}\right) \\
    v_c &= \sqrt{2} V_s \cos \left(\omega_s t + \frac{2\pi}{3}\right) = \sqrt{2} v_d \cos \left(\theta_s + \frac{2\pi}{3}\right) - \sqrt{2} v_q \sin \left(\theta_s + \frac{2\pi}{3}\right)
\end{align*}

where the $dq$ stator voltages $(v_d, v_q)$ are scaled to have the same magnitude as the RMS values of the stator phase voltages $(V_s)$. From (5.15), we notice that by aligning the $d$-axis of the rotating reference frame with the stator voltage vector position, the $q$-component of the stator voltage is equal to zero. The amplitude of the $d$-component is equal to the RMS value of the stator phase voltage and, under balanced network conditions, is constant. Reference frame transformations are described in more detail in Appendix B.

If the losses of the three-phase line connecting the converter with the grid are neglected ($R = 0$), the RMS phase values of the stator voltages $(V_s)$ and the converter voltages $(V_{si})$ are equal. By taking into account the stator-side converter voltage transfer characteristic (5.7), we can write

$$
V_s = v_d = \frac{P_m}{2\sqrt{2}} \cdot v_{DC}
$$

The power-balance equation for the stator-side converter can be written in $dq$ coordinates from (5.10), with the appropriate scaling of the $dq$ variables, as

$$
P_{conv} = v_{DC}i_{DC} = 3v_di_d
$$

By combining (5.16) and (5.17), we obtain a direct relationship between the DC
stator-side converter current \( i_{DCs} \) and the \( d \)-axis stator-side converter current \( i_{d1} \), as

\[
i_{DCs} = \frac{3}{2\sqrt{2}} P_m i_{d1}
\]

From Kirchoff's 1st law for the DC link currents of the voltage converter, we obtain the following expression, with the resistive losses neglected,

\[
C \frac{dv_{DC}}{dt} = i_{DCs} - i_{DCr} = \frac{3}{2\sqrt{2}} P_m i_{d1} - i_{DCr}
\]

where \( C \) is the value of a DC-link capacitor as depicted in Fig. 5.3. Hence the DC-link voltage \( v_{DC} \) can be controlled via \( i_{d1} \).

### 5.6.2 Stator-side Converter Control

The objective of the stator-side converter control is to keep the DC-link voltage constant, regardless of the magnitude and direction of the rotor power. A vector-control approach is used with the reference frame oriented along the stator-voltage vector position, enabling independent control of active and reactive power flowing between the stator and the stator-side converter. The stator-side PWM converter is current regulated, with the direct-axis current component \( i_{d1} \) used to regulate the DC-link voltage and the quadrature-axis current component \( i_{q1} \) used to regulate the reactive power.

![Stator-side converter controller](image)

Figure 5.5: Stator-side converter controller

The first stage of the stator-side converter controller is a current controller that
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

contains two current control loops, for the \( d \)- and \( q \)-axis current components. The reference of the direct-axis current component is set by a slower DC voltage-control loop (Stage 2), as depicted in Fig. 5.5. The reference of the quadrature-axis current component corresponding to the reactive power can be used for optimum power sharing between the generator and the grid-side converter or kept to a constant value.

The outputs of current controllers \( v'_d \) and \( v'_q \) represent the regulated voltage drops across the stator-side connecting line \((L,R)\), which is used to calculate the reference values \( v^*_{d1} \) and \( v^*_{q1} \) of the stator-side converter. Equation (5.9) can be written in the \( dq \) reference frame as:

\[
\begin{align*}
    v_d &= R_i d_1 + L \frac{d}{dt} d_1 - \omega_s L q_1 + v_{d1} \\
    v_q &= R_i q_1 + L \frac{d}{dt} q_1 + \omega_s L d_1 + v_{q1}
\end{align*}
\]

Taking into account that \( v_q = 0 \) in the stator-voltage-oriented \( dq \) reference frame, reference voltages of the stator-side converter are calculated from (5.20) as:

\[
\begin{align*}
    v^*_{d1} &= -v'_d + (\omega_s L q_1 + v_d) \\
    v^*_{q1} &= -v'_q - (\omega_s L d_1)
\end{align*}
\]

Finally, the reference voltages are converted back to phase domain using the inverse \( dq \) transformation, as follows:

\[
\begin{bmatrix}
    v^*_a \\
    v^*_b \\
    v^*_c
\end{bmatrix} = \sqrt{2} \begin{bmatrix}
    \cos \theta_s & -\sin \theta_s \\
    \cos (\theta_s - \frac{2\pi}{3}) & -\sin (\theta_s - \frac{2\pi}{3}) \\
    \cos (\theta_s + \frac{2\pi}{3}) & -\sin (\theta_s + \frac{2\pi}{3})
\end{bmatrix} \begin{bmatrix}
    v^*_{d1} \\
    v^*_{q1}
\end{bmatrix}
\]

where the multiplier \( \sqrt{2} \) accounts for the scaling of the \( dq \) variables to RMS phase values.

5.6.3 Rotor-side Converter Model

The rotor-side converter is connected directly to the rotor terminals of a DFIG. Therefore, the rotor-side converter AC terminal voltages are the voltages across the rotor windings. The power conservation equation between the DC link and the rotor-side converter, with the losses neglected, can be written similarly to (5.10) as:

\[
P_{\text{conv}} = v_A i_A + v_B i_B + v_C i_C = v_{DC} i_{DCr}
\]

where all voltages and currents are instantaneous phase values.

The rotor-side converter is vector controlled in the stator flux-oriented \( dq \) reference
frame. The angular position of the stator-flux vector $\Psi_s$ can be calculated from
the stationary $\alpha\beta$ stator-flux components in an equivalent way as presented for
the stator-voltage vector angular position in Section 5.6.1. However, in this case the
instantaneous values of the stator-flux linkages $\psi_a, \psi_b$, and $\psi_c$ (calculated from (4.21))
have to be filtered out using a digital pass-band filter to eliminate DC offsets. Filtered
values are denoted as $\psi_{a,F}, \psi_{b,F}$ and $\psi_{c,F}$ in Fig. 5.6, depicting relationships between
stator-flux linkage vectors in different reference frames. Note that by aligning the
d-axis with the stator-flux vector position, the $q$-component of the stator-flux vector
is equal to zero. Digital filtering of instantaneous stator-flux linkages is described in
Appendix C.

Figure 5.6: Geometric representation of the relationship between stationary $abc$ and
$\alpha\beta$ reference-frame flux linkages, and definition of stator-flux-oriented $dq$ reference
frame

The angular position of the stator-flux vector $\theta_{sf}$ is obtained from the instantaneous
values of the filtered stator-flux linkages as

$$\theta_{sf} = \tan^{-1}\left(\frac{\sqrt{3}\psi_{b,F} - \sqrt{3}\psi_{c,F}}{\frac{\sqrt{3}}{2}\psi_{a,F} - \frac{1}{2}\psi_{b,F} - \frac{1}{2}\psi_{c,F}}\right)$$  \hspace{1cm} (5.24)
The transformation of the rotor-phase variables into a rotating \(dq\) reference frame, where the \(dq\) variables are scaled to have the same amplitude as RMS values of phase variables, can be written for the rotor voltages as

\[
\begin{bmatrix}
v_{dr} \\
v_{qr}
\end{bmatrix} = \frac{1}{\sqrt{2}} \cdot \frac{2}{3} \begin{bmatrix}
\cos(\theta_{slip}) & \cos(\theta_{slip} - \frac{2\pi}{3}) & \cos(\theta_{slip} + \frac{2\pi}{3}) \\
-\sin(\theta_{slip}) & -\sin(\theta_{slip} - \frac{2\pi}{3}) & -\sin(\theta_{slip} + \frac{2\pi}{3})
\end{bmatrix}
\begin{bmatrix}
v_A \\
v_B \\
v_C
\end{bmatrix}
\] (5.25)

where \(\theta_{slip} = \omega_{sf} - \omega_r\) is the slip’s angular velocity and \(\omega_r\) is the rotor’s angular position, defined in Section 4.4.1. The transformation is valid for the rotor voltages and currents as well.

Equations relating rotor voltages in the \(abc\) phase domain and the \(dq\) rotating reference frame can be written using the inverse \(dq\) transformation as

\[
v_A = \sqrt{2}V_r \cos(\omega_{slip} t + \varphi) = \sqrt{2}v_{dr} \cos(\theta_{slip}) - \sqrt{2}v_{qr} \sin(\theta_{slip})
\]

\[
v_B = \sqrt{2}V_r \cos(\omega_{slip} t + \varphi - \frac{2\pi}{3}) = \sqrt{2}v_{dr} \cos\left(\frac{2\pi}{3}\right) - \sqrt{2}v_{qr} \sin\left(\frac{2\pi}{3}\right)
\]

\[
v_C = \sqrt{2}V_r \cos(\omega_{slip} t + \varphi + \frac{2\pi}{3}) = \sqrt{2}v_{dr} \cos\left(\frac{2\pi}{3}\right) - \sqrt{2}v_{qr} \sin\left(\frac{2\pi}{3}\right)
\] (5.26)

where \(\omega_{slip} = \omega_{sf} - \omega_r\) is the relative angular velocity between the rotor and the reference \(dq\) axes. The magnitude and phase angle of the rotor-voltage vector with respect to the rotating \(dq\) reference frame are defined as

\[
V_r = \sqrt{v_{dr}^2 + v_{qr}^2}
\]

\[
\varphi = \tan^{-1} \frac{v_{qr}}{v_{dr}}
\] (5.27)

Similar to its presentation for the stator-side converter model, the rotor-side converter voltage transfer characteristic can be written as

\[
V_r = \frac{P_{mr}}{2\sqrt{2}}v_{DC}
\] (5.28)

where \(P_{mr}\) the modulation depth of the rotor-side converter. Since the rotor-side converter is controlled to the desired value in the stator-flux-oriented \(dq\) reference frame, it is convenient to decompose the modulation depth into \(dq\) coordinates, to
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

obtain the following expressions:

\[
\begin{align*}
    v_{dr} &= \frac{P_{mrd}}{2\sqrt{2}} v_{DC} \\
    v_{qr} &= \frac{P_{mrq}}{2\sqrt{2}} v_{DC}
\end{align*}
\]  

(5.29)

The power-balance equation for the rotor-side converter (5.23) can be written in \(dq\) variables with appropriate scaling as

\[
P_{\text{conv}} = v_{DC} i_{DCr} = 3 (v_{dr} i_{dr} + v_{qr} i_{qr})
\]

(5.30)

By combining (5.29) and (5.30), we obtain the following expression:

\[
i_{DCr} = \frac{3}{2\sqrt{2}} (P_{mrd} i_{dr} + P_{mrq} i_{qr})
\]

(5.31)

which, in combination with (5.19), gives a differential equation describing the DC link of the voltage converter in \(dq\) coordinates as

\[
C \frac{dv_{DC}}{dt} = \frac{3}{2\sqrt{2}} P_m i_{d1} - \frac{3}{2\sqrt{2}} (P_{mrd} i_{dr} + P_{mrq} i_{qr})
\]

(5.32)

5.6.4 Rotor-side Converter Control

The PWM converter inserted in the rotor circuit allows for fast and flexible control of the DFIG by modifying the magnitude and phase angle of the rotor voltage. As mentioned in the previous section, the induction machine is controlled in a synchronously rotating \(dq\) reference frame with the \(d\)-axis aligned along the stator-flux vector position. In this way, decoupled control between the electrical torque and the reactive power is obtained.

The rotor-side converter is controlled by a two-stage controller. Stage 1 consists of very fast current controllers regulating the rotor \(d\)-axis and \(q\)-axis currents to their reference values. The reference of the quadrature-axis current component is specified by a slower speed controller (Stage 2) as shown in Fig. 5.7. The \(q\) component of the rotor current directly influences the torque, so \(i_{qr}\) can be used for torque or active power control. The \(d\)-axis component is a reactive current component, so \(i_{dr}\) can be used for reactive power or voltage control. Assuming that all reactive power is supplied by the stator, \(i_{dr}^*\) may be set to zero.

The outputs of the current controllers \(v'_{dr}\) and \(v'_{qr}\), in a similar way to the stator-side converter controller, are used to calculate the reference values of \(v_{dr}^*\) and \(v_{qr}^*\) of
the rotor-side converter. The voltage equations of the rotor circuit in $dq$ reference frame can be written as

$$
\begin{align*}
    v_{dr} &= R_r i_{dr} + \left( L_r - \frac{L_o^2}{L_s^{(ref)}} \right) \frac{di_{dr}}{dt} - \omega_{\text{slip}} \left( L_r - \frac{L_o^2}{L_s^{(ref)}} \right) i_{qr} \\
    v_{qr} &= R_r i_{qr} + \left( L_r - \frac{L_o^2}{L_s^{(ref)}} \right) \frac{di_{qr}}{dt} + \omega_{\text{slip}} \left( \frac{L_o^2}{L_s^{(ref)}} i_m^{(ref)} + \left( L_r - \frac{L_o^2}{L_s^{(ref)}} \right) i_{dr} \right)
\end{align*}
$$

(5.33)

where $R_r$, $L_r$, $L_o$ and $L_s^{(ref)}$ are rotor-referred quantities related to per-phase resistances and reactances of an induction machine (Section 4.4.1.1) as

$$
\begin{align*}
    R_r &= R_a = R_b = R_c \\
    L_r &= L_{Lr} + \frac{3}{2} L_{mr} \\
    L_o &= \frac{3}{2} L_{mr} \\
    L_s^{(ref)} &= \left( L_{ls} + \frac{3}{2} L_{ms} \right) \left( \frac{N_r}{N_s} \right)^2
\end{align*}
$$

(5.34)

and $i_m^{(ref)}$ is the rotor-referred RMS value of the stator magnetizing current. $i_m^{(ref)}$ can be calculated from the following expression:

$$
    i_m^{(ref)} = \frac{V_s}{\omega_s L_o} \left( \frac{N_r}{N_s} \right)
$$

(5.35)
Equation (5.33) takes into account that in the stator-flux-oriented reference frame, the \(dq\) components of the stator flux are \(\psi_{sd} = \psi_s = L_0 i_{ms} \) and \(\psi_{sq} = 0\). The reference voltages of the rotor-side converter are calculated from (5.33) as

\[
\begin{align*}
    v_{dr}^* &= v_{dr}' - \omega_{\text{slip}} \left( L_r - \frac{L_o^2}{L_s^{(\text{ref})}} \right) i_{qr} \\
    v_{qr}^* &= v_{qr}' + \omega_{\text{slip}} \left( \frac{L_o^2}{L_s^{(\text{ref})}} i_{ms} + \left( L_r - \frac{L_o^2}{L_s^{(\text{ref})}} \right) i_{dr} \right) 
\end{align*}
\] (5.36)

The reference voltages are converted back to phase domain using the inverse \(dq\) transformation with appropriate scaling as follows:

\[
\begin{bmatrix}
    v_A^* \\
    v_B^* \\
    v_C^*
\end{bmatrix}
= \sqrt{2} \begin{bmatrix}
    \cos \theta_{\text{slip}} & -\sin \theta_{\text{slip}} \\
    \cos \left( \theta_{\text{slip}} - \frac{2\pi}{3} \right) & -\sin \left( \theta_{\text{slip}} - \frac{2\pi}{3} \right) \\
    \cos \left( \theta_{\text{slip}} + \frac{2\pi}{3} \right) & -\sin \left( \theta_{\text{slip}} + \frac{2\pi}{3} \right)
\end{bmatrix}
\begin{bmatrix}
    v_{dr}^* \\
    v_{qr}^*
\end{bmatrix}
\] (5.37)

### 5.7 Wind Turbine Model

The kinetic energy of a mass of air \(m\) of velocity \(v_{\text{wind}}\) is given by a well-known equation:

\[
E_k = \frac{m}{2} \cdot v_{\text{wind}}^2
\] (5.38)

The power of the moving air mass is the derivative of the kinetic energy with respect to time, and can be written as

\[
P_0 = \frac{\partial E_k}{\partial t} = \frac{1}{2} \cdot \frac{\partial m}{\partial t} \cdot v_{\text{wind}}^2 = \frac{1}{2} \cdot q \cdot v_{\text{wind}}^2
\] (5.39)

where \(q\) represents the mass flow given by the following expression:

\[
q = \rho \cdot v_{\text{wind}} \cdot A
\] (5.40)

where

- \(A\) is a cross section of the air flow
- \(\rho\) is air density

Only a fraction of the wind kinetic energy can be converted into rotational power at the wind turbine shaft. This fraction of power \(P_{\text{wind}}\) depends on the wind velocity, rotor speed and blade position (if the turbine is pitch or active-stall controlled) and on the turbine design. It is usually quantified by a constant \(C_p\), representing aerodynamic
efficiency

\[ C_p = \frac{P_{\text{wind}}}{P_0} \]  

(5.41)

For a specific turbine design, \( C_p \) is normally calculated as a function of the pitch-angle \( \beta \) and the tip-speed ratio \( \lambda \). The tip-speed ratio is given as

\[ \lambda = \frac{\omega_t \cdot R}{v_{\text{wind}}} \]  

(5.42)

where \( R \) is the radius of the turbine blades and \( \omega_t \) is the turbine speed. \( C_p(\lambda, \beta) \) is normally defined in the form of a two-dimensional lookup characteristic for different values of \( \beta \) and \( \lambda \). Alternatively, analytical approaches for approximating the \( C_p \) characteristic can be used, such as the following expression [39]:

\[ C_p = (0.44 - 0.0167\beta) \sin \frac{\pi(\lambda - 2)}{13 - 0.3\beta} - 0.00184(\lambda - 2)\beta \]  

(5.43)

The mechanical power extracted from the wind (with \( A = \pi R^2 \)) is calculated from

\[ P_{\text{wind}} = \frac{1}{2} \cdot \rho \cdot \pi \cdot R^2 \cdot C_p(\lambda, \beta) \cdot v_{\text{wind}}^3 \]  

(5.44)

with the mechanical torque driving the induction generator calculated from (5.6).

5.8 Maximum Power Tracking

The family of \( P_{\text{wind}} - \omega_r \) curves can be derived from (5.44) for various values of wind velocity, with \( \omega_r \) representing the shaft speed referred to the generator side of the gearbox. For the 7.5 kW wind turbine of [36], the curves are derived for fixed \( \beta \), as shown in Fig. 5.8. Curve \( P_{\text{opt}} \) defines the maximum energy captured from the wind, and is characterized by the factor \( K_{\text{opt}} \) in the following expression:

\[ P_{\text{opt}} = K_{\text{opt}} \omega_r^3 \]  

(5.45)

The objective of the tracking control is to keep the turbine speed fixed on the \( P_{\text{opt}} \) curve as the wind velocity varies. For wind velocity higher than the turbine's rating, the turbine energy captured has to be limited by applying pitch control or driving the machine to the stall point.

One method of maximum power tracking is realized by observing the mechanical torque on the generator shaft, and driving the DFIG to the optimum power curve by
setting the reference rotor speed of the rotor-side converter controller (Fig. 5.7) to

\[ \omega_r^* = \sqrt{\frac{T_m}{K_{opt}}} \]  

(5.46)

This method of maximum power tracking is referred to as a speed-mode control. For extracted wind power higher than rated, the DFIG is driven to a stall point for the particular wind velocity, where

\[ \omega_r^* = \frac{P_{max}}{T_m} \]  

(5.47)

In Fig. 5.8 the rated power of \( P_{max} = 7.5 \) kW defines the stall point for wind velocities higher than the rated (10 m/s).

![Figure 5.8: Wind turbine characteristics with maximum energy capture curve](image)

Another method of maximum power tracking is a current-mode control where for the given shaft speed an electromagnetic torque is imposed on the DFIG (after compensating for friction losses) given from:

\[ T_e^* = K_{opt}\omega_r^2 - B\omega_r \]  

(5.48)

by regulating the q-axis rotor current \( i_{qr}^* \) to

\[ i_{qr}^* = -\frac{2I_s^{(ref)} T_e^*}{3p_I L_i^{(ref)} L_{ms}^2} \]  

(5.49)
The choice of a speed-mode or current-mode control is made by choosing the desired position of a switch in the rotor-side converter controller (Fig. 5.7).

### 5.9 Implementation of a DFIG Wind Turbine System in OVNI

The doubly fed induction generator wind turbine system model consists of the phase-domain induction generator discrete model described in Section 4.4.1.3, the discrete phase-domain model of the voltage converter and the voltage-converter controllers operating in the dq reference frame.

The phase-domain voltage converter model is composed of the voltage transfer characteristics of the stator and rotor-side converters plus the differential equation of the DC link. The required values of the converter’s terminal voltages are calculated as follows:

\[
\begin{align*}
    v_{a1}^* &= \frac{P_{\text{md}1}^*}{2} v_{DC} \cos \theta_s - \frac{P_{\text{mq}1}^*}{2} v_{DC} \sin \theta_s \\
    v_{b1}^* &= \frac{P_{\text{md}1}^*}{2} v_{DC} \cos \left( \theta_s - \frac{2\pi}{3} \right) - \frac{P_{\text{mq}1}^*}{2} v_{DC} \sin \left( \theta_s - \frac{2\pi}{3} \right) \\
    v_{c1}^* &= \frac{P_{\text{md}1}^*}{2} v_{DC} \cos \left( \theta_s + \frac{2\pi}{3} \right) - \frac{P_{\text{mq}1}^*}{2} v_{DC} \sin \left( \theta_s + \frac{2\pi}{3} \right) \\
    v_A^* &= \frac{P_{\text{md}r}^*}{2} v_{DC} \cos \theta_{\text{slip}} - \frac{P_{\text{mq}r}^*}{2} v_{DC} \sin \theta_{\text{slip}} \\
    v_B^* &= \frac{P_{\text{md}r}^*}{2} v_{DC} \cos \left( \theta_{\text{slip}} - \frac{2\pi}{3} \right) - \frac{P_{\text{mq}r}^*}{2} v_{DC} \sin \left( \theta_{\text{slip}} - \frac{2\pi}{3} \right) \\
    v_C^* &= \frac{P_{\text{md}r}^*}{2} v_{DC} \cos \left( \theta_{\text{slip}} + \frac{2\pi}{3} \right) - \frac{P_{\text{mq}r}^*}{2} v_{DC} \sin \left( \theta_{\text{slip}} + \frac{2\pi}{3} \right)
\end{align*}
\]

where \( P_{\text{md}1}^* \), \( P_{\text{mq}1}^* \), \( P_{\text{md}r}^* \) and \( P_{\text{mq}r}^* \) are control variables of the voltage converter defined by (5.16), (5.28).
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

The DC link equation in the $abc$ reference frame has the following form:

$$C \frac{dv_{DC}}{dt} = i_a \left( \frac{P_{md1}^*}{2} \cos \theta_s - \frac{P_{mq1}^*}{2} \sin \theta_s \right)$$

$$+ i_b \left( \frac{P_{md1}^*}{2} \cos \left( \theta_s - \frac{2\pi}{3} \right) - \frac{P_{mq1}^*}{2} \sin \left( \theta_s - \frac{2\pi}{3} \right) \right)$$

$$+ i_c \left( \frac{P_{md1}^*}{2} \cos \left( \theta_s + \frac{2\pi}{3} \right) - \frac{P_{mq1}^*}{2} \sin \left( \theta_s + \frac{2\pi}{3} \right) \right)$$

$$- i_A \left( \frac{P_{mdr}^*}{2} \cos \theta_{slip} - \frac{P_{mqr}^*}{2} \sin \theta_{slip} \right)$$

$$- i_B \left( \frac{P_{mdr}^*}{2} \cos \left( \theta_{slip} - \frac{2\pi}{3} \right) - \frac{P_{mqr}^*}{2} \sin \left( \theta_{slip} - \frac{2\pi}{3} \right) \right)$$

$$- i_C \left( \frac{P_{mdr}^*}{2} \cos \left( \theta_{slip} + \frac{2\pi}{3} \right) - \frac{P_{mqr}^*}{2} \sin \left( \theta_{slip} + \frac{2\pi}{3} \right) \right)$$

(5.51)

and is discretized using the trapezoidal integration rule. These equations are combined with the equations describing the phase-domain induction machine presented in Section 4.4.1.

5.10 DFIG Wind Turbine Test System

In this thesis, we have modelled a doubly fed induction generator wind turbine system from the literature using the Multilevel MATE concept, and implemented it in OVNI. Reference [36] describes a doubly fed induction generator application to variable-speed wind-energy generation using back-to-back PWM converters in an experimental setup. The paper provides details on the control setup of the stator- and rotor-side converters and provides the data necessary for modelling of the wind turbine system in the phase domain. Modelling details have been presented in the preceding sections of this chapter.

We extended the DFIG wind turbine system case to include a step-up transformer and a 10 kV double-circuit transmission line connecting the wind turbine to a strong network modelled as an infinite bus. The single-line diagram of the test network is depicted in Fig. 5.9. For the purpose of comparison, the wind turbine generator defined in Appendix D is connected to a 250 V bus to replicate the situation in [36]. In addition, the inclusion of the connecting network allows us to study the response of the wind turbine to a three-phase fault applied in the middle of Circuit 2 of the 10 kV, 10 km transmission line. We obtained the data for modelling the step-up transformer and the transmission line from typical values [40].

In the following section, simulation results are presented showing the response of
the DFIG wind turbine system to a change in wind velocity following the maximum speed tracking system described in Section 5.8. Subsequently, we show the responses of a DFIG system to a three-phase fault, and compare them against the results calculated with the Transient Stability Assessment Tool TSAT [41].

5.11 Simulation Results

5.11.1 Transient Response of the DFIG Wind Turbine to a Step Decrease in Wind Velocity

Initially, the induction machine is operating in steady state with a wind velocity of 9 m/s, corresponding to the optimal shaft speed of 1350 rpm and mechanical torque of 38 Nm. The maximum power captured at this wind speed is 5400 W. The maximum power-tracking system works in current-mode control.

At $t = 2\ s$, the wind velocity decreases instantaneously to 5 m/s, which corresponds to the optimal shaft speed of 750 rpm and mechanical torque of 12 Nm. The maximum power captured from the wind at this speed is 932 W. Reduced wind power, and therefore mechanical torque, cause the DFIG to decelerate, with the deceleration torque being the difference between the turbine's mechanical torque and the torque given by the optimum curve (5.48). A similar situation with opposite effects would occur in the case of an increase in wind velocity. This scenario corresponds to the scenario described in [36] for the experimental setup. By simulating the response of the DFIG model built with OVNI to the same type of disturbance, a comparison of the results can be obtained and conclusions can be drawn about the accuracy of the modelling approach.
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

In current-mode control, the DFIG reaches the new steady state after about sixty seconds. The DFIG is initially running above the synchronous speed of 1000 rpm (rated frequency of the system is 50 Hz) for $v_{\text{wind}} = 9 \text{ m/s}$, and in the new steady state it runs below the synchronous speed for $v_{\text{wind}} = 5 \text{ m/s}$. The electromagnetic torque corresponding to the mechanical torque on the shaft after compensating for the friction losses decreases from the initial 30 Nm to around 7.5 Nm, corresponding to the new steady state with a wind velocity of 5 m/s. A slight decrease of the stator flux reflects the decrease in voltage at the stator terminals.

Figure 5.10: Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) rotor angular velocity, (b) electromagnetic torque, (c) rotor speed, (d) stator flux
Initially, for a wind velocity of 9 m/s, the DFIG generates around 3800 W of active power and consumes around 3100 VAr of reactive power. Most of the active power of the DFIG is being delivered by the stator (3100 W), and the rest is delivered by the rotor via the converter. For the new steady state with wind speed of 5 m/s, the DFIG generates 500 W of active power and consumes about 200 VAr less of reactive power. In this latter case, the stator side delivers 720 W of active power and the rotor side consumes 220 W. We demonstrate here the capability of the DFIG to deliver active power both via the stator and the rotor of the machine.
Figure 5.12: Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator-side converter active power, (b) stator-side converter reactive power, (c) rotor-side converter (machine rotor) active power, (d) rotor-side converter (machine rotor) reactive power

In Fig. 5.12(a), we notice the change of the stator-side converter active power from positive (generating) to negative (consuming) as the rotor speed moves from supersynchronous to subsynchronous due to the decrease in wind velocity. The stator-side converter active power depicted in Fig. 5.12(a) is equal to the rotor-side converter active power in Fig. 5.12(c) (with the converter losses neglected). The point where the active power of the converter is zero corresponds to the point where the rotor speed is equal to the synchronous speed of the induction generator. The reactive power of the stator-side converter is regulated to zero by the stator-side converter control ($i_{q1} = 0$).
A decrease in the DFIG power output causes a decrease in the RMS values of the electrical variables at the DFIG terminals. For the RMS rotor voltage value depicted in Fig. 5.13(c), the minimum value corresponds to the time instant when the rotor speed is equal to the synchronous speed.
Figure 5.14: Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator phase to neutral voltage, (b) stator phase current, (c) rotor phase to neutral voltage, (d) rotor phase current, (e) stator-side converter phase voltage, (f) stator-side converter phase current.

Figure 5.14 depicts the DFIG currents and voltages in the phase domain. Figures 5.14(c) and Fig. 5.14(d) show the smooth operation of the DFIG through synchronous speed. By examining Fig. 5.14(f) we note the change of “direction” of the stator-side converter current, indicating the change in converter power from generation to consumption.
Figure 5.15: Transient response of the DFIG wind turbine to a step decrease in wind velocity: (a) stator voltages $dq$ components, (b) stator currents $dq$ components, (c) rotor voltages $dq$ components, (d) rotor currents $dq$ components, (e) stator-side converter voltages $dq$ components, (f) stator-side converter currents $dq$ components.

Figure 5.15 shows the $dq$ components of the DFIG electrical quantities. Stator quantities are shown in the stator voltage-oriented reference frame, and rotor quantities are shown in the stator flux-oriented reference frame. The control of the $q$-axis stator-side converter current $i_{q1}$ determines the displacement factor at the DFIG terminals, and it is kept at zero in this case (no reactive power is supplied to the network by the converter). For supersynchronous operation of the DFIG, this implies a phase displacement of $180^\circ$ between the stator-side converter phase voltage and current, and for subsynchronous operation it gives a unity displacement factor. The control of the $d$-axis stator-side converter current $i_{d1}$ determines the active power exchanged with the network for the purpose of maintaining the converter DC-link voltage at a
constant value.

The induction machine is controlled through rotor-side control in the $dq$ reference frame, with the d-axis current component controlling the machine's reactive power, and the q-axis current component being proportional to the machine's electrical torque. Assuming that all reactive power is supplied to the machine by the stator, the d-axis rotor current is set to zero. By controlling the q-axis rotor current a reference torque or a reference speed can be imposed on the DFIG.

![Graphs showing comparisons](image)

**Figure 5.16:** Comparison of a transient response of the DFIG wind turbine to a step decrease in wind velocity against the experimental results from the literature: (a) rotor speed, (b) d-axis rotor current, (c) q-axis rotor current

Our simulation results compared against the experimental results published in [36] for a decrease of wind velocity are shown in Fig. 5.16. It can be seen from the figure that the results showed good agreement. Differences can be attributed to a number of factors, such as using an average model to model the PWM converter, neglecting saturation, neglecting details of the machine design contributing to the generator winding/slot harmonics, etc.
Chapter 5. Modelling and Simulation of a DFIG Wind Turbine System in OVNI

5.11.2 Transient Response of the DFIG Wind Turbine to a Three-phase Short Circuit

Initially, the induction machine operates in steady state with a wind velocity of 8 m/s, generating 2700 W of active power that is transmitted to a “strong” power system modelled as an infinite bus. The DFIG stator-side converter is regulated to maintain its q-axis current at zero, meaning that the reactive power of the induction generator is entirely supplied from the network. At \( t = 0.2 \) s, a three-phase fault is applied in the middle of circuit CCT2, as shown in Fig. 5.9. The fault is removed after 0.1 s without tripping the circuit. The response of the wind turbine system to the fault is depicted in Fig. 5.17 - Fig. 5.24.

Large disturbances cause large initial fault currents both in the stator and rotor. Moreover the DC link will experience overvoltages due to reduced voltage at the stator-side converter terminals. High currents flowing through the converter may activate the crowbar protection in order to preserve the converter which may then lead to the wind turbine being disconnected from the network. It has been recognized in the literature that stability models of DFIG wind turbines may not adequately estimate the transients in the rotor currents [42], therefore they may not provide correct responses to system disturbances. To verify this statement, a transient response of a DFIG to a three-phase fault has been simulated with the Transient Security Assessment Tool (TSAT) [41], which is a good representative of commonly used stability simulation tools.

The exact same case as tested with OVNI could not be replicated due to the design of TSAT [41], which allows accurate computations in the MVA range rather than in the kVA range. Therefore the machine and the network parameters were scaled to reflect the power flow at the MVA level. In particular, the DFIG machine generating 2.7 kW was scaled to generate 2.7 MW of active power and consume 3 MVAr of reactive power. Default controls of a DFIG in TSAT were modified to reflect similar conditions as in the OVNI model.

The DFIG model used in TSAT is a user-defined model of type 3 that represents a rotor voltage-controlled DFIG with flux transients. The shape and general trends of all the responses computed with TSAT correspond to those calculated with OVNI. The most significant difference in the results is found in the oscillatory behavior caused by the transient in the converter’s DC link. As is commonly done in transient stability modelling, the stator-side converter and the DC link are neglected and only the rotor-side converter control is modelled. By examining the results shown in Figs. 5.25 and 5.26, one can conclude that this is probably an oversimplification imposed on the model. This type of behavior, which is not well represented with stability models, is crucial for determining the correct responses of protective devices, such as the aforementioned crowbar protection. A more interesting comparison could be made between the stability model that includes the stator-side converter model and the
detailed model described in this work, both implemented in OVNI. This task, however, is left for future work.

5.11.3 Commentary on the Comparison of Transient Simulation Results Obtained with EMTP-type and Stability-type Tools

The preceding section compares transient simulation results obtained with OVNI, which is an EMTP-type of tool, and TSAT, which represents a typical stability simulation tool. A similar comparison between any other EMTP- and stability-type tool will have a similar outcome. It is very well-known that EMTP-type of modelling is more accurate than stability modelling.

Smaller scale systems, such as the DFIG WTG tested in this work, can be successfully replicated in a few commercially available EMTP-type simulators. However, these simulators do not implement partitioning techniques. Our OVNI simulator implements the concept of MATE (or, in the more general case, Multilevel MATE described in Chapter 3), which allows parallel processing and efficient computation at the subsystem level. Multilevel MATE can be applied to simulations of very large power system networks. Stability tools use approximate modelling to cope with the size of the network, while OVNI uses multilevel system partitioning and efficient modelling solutions to achieve this objective. Therefore, for the same objective of coping with system size, Multilevel MATE allows more detailed modelling than do the stability tools. The solutions proposed in this thesis are intended to make OVNI a general purpose simulator capable of performing EMTP and stability simulations of large power systems. Other EMTP-type tools are not oriented toward this purpose.

5.12 Summary

In this chapter, a detailed example of modelling of a doubly fed induction generator wind turbine system is presented. Using our simulation tool we have replicated and simulated an experimental setup from the literature. Our comparison of responses to a step change in wind speed has shown good agreement with the experimental results. In addition, we have the same system simulated for a three-phase fault near the DFIG terminals. We have also made a qualitative comparison to a similar case setup using a commercially available transient stability simulation tool.
Figure 5.17: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) rotor angular velocity, (b) electromagnetic torque, (c) rotor speed, (d) stator flux
Figure 5.18: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) DFIG active power, (b) DFIG reactive power, (c) machine stator active power, (d) machine stator reactive power
Figure 5.19: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator-side converter active power, (b) stator-side converter reactive power, (c) rotor-side converter (machine rotor) active power, (d) rotor-side converter (machine rotor) reactive power.
Figure 5.20: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) DFIG stator-side RMS voltage, (b) DFIG stator-side RMS current, (c) DFIG rotor-side RMS voltage, (d) DFIG rotor-side RMS current
Figure 5.21: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator phase to neutral voltage, (b) stator phase current, (c) rotor phase to neutral voltage, (d) rotor phase current, (e) stator-side converter phase voltage, (f) stator-side converter phase current
Figure 5.22: Transient response of the DFIG wind turbine DC-link voltage to a three-phase short circuit
Figure 5.23: Transient response of the DFIG wind turbine to a three-phase short circuit: (a) stator voltages' dq components, (b) stator currents' dq components, (c) rotor voltages' dq components, (d) rotor currents' dq components, (e) stator-side converter voltages' dq components, (f) stator-side converter currents' dq components.
Figure 5.24: Transient response of the DFIG wind turbine to a three-phase short circuit: rotor phase voltages and currents
Figure 5.25: Transient response of the DFIG wind turbine to a three-phase short circuit obtained with the stability tool TSAT: (a) DFIG stator terminal voltage, (b) DFIG rotor angular velocity, (c) DFIG active power, (d) DFIG reactive power
Figure 5.26: Transient response of the DFIG wind turbine to a three-phase short circuit obtained with the stability tool TSAT: (a) DFIG rotor dq voltages and currents (from top to bottom - $v_{dr}$, $i_{dr}$, $v_{qr}$ and $i_{qr}$), (b) DFIG stator current
Chapter 6

Conclusion

This chapter summarizes the main contributions of the work presented in the preceding chapters and provides suggestions for the continuation of this research.

6.1 Summary of Contributions

This thesis makes the following contributions to the field of power system transient simulations:

• Development of the new Multilevel MATE Concept.

We have presented a new concept of multilevel system partitioning and functional modelling for the simulation of power system networks. The proposed concept by its nature allows a more efficient solution of the subsystems' equations, and provides a means for general modelling of power system components with nodal and branch equations.

The advantages of subsystem partitioning with the Multilevel MATE concept introduced in this work are presented in terms of the computational speedup over the existing single-level MATE. The greatest improvements are realized when simulating subsystems containing components of a changing nature. The advantages of modelling with branch equations are demonstrated through the implementations of an ideal switch and a nonlinear control function. With the Multilevel MATE concept, the subsystem's admittance matrix is not affected by a changing topology due to switching, and its order is not increased by adding branch equations. With this approach, branch currents can be calculated at every time step or only when needed.

Nonlinear functions of state variables of the system require solutions with iterations at every time step of the simulation. With our new approach, the nonlinear equations iterate against the linear systems' Thévenin equivalents alone, which, in the MATE approach, are calculated at the level of the links.

• Modelling of Power System Components with Multilevel MATE. Multilevel MATE provides a very general approach to modelling of voltage sources, grounded or ungrounded, by means of sublinks. A solution for conditioning
Chapter 6. Conclusion

of near-singular or singular nodal equations is provided by the introduction of compensating shunt impedances. Also, the solution for calculating ungrounded subsystem reference voltages is given.

We describe branch (sublink) implementation of phase-domain induction and synchronous machine models with Multilevel MATE. The advantage of branch implementation is evident from the fact that the changing nature of the machine's coupled branch equations does not affect the subsystem's admittance matrix. Also, the transformation of coupled branch equations into nodal equations is no longer required, which greatly simplifies the modelling approach. Multilevel MATE enables the controller's equations (linear and nonlinear) to be conveniently incorporated. Depending on the scope of the controller, global and/or local control is easily implemented by means of links and sublinks.

- Validation of Modelling with Multilevel MATE through an Example of a Doubly-Fed Induction Generator Wind Turbine System. We test our proposed approach to modelling and simulation on an example of a doubly fed induction generator wind turbine system. We describe modelling of the wind turbine system components and control in detail. We have replicated an experimental setup from the literature [36] in our simulation tool and tested it for two types of disturbances: decrease in wind velocity and a three-phase fault in the connecting double-circuit transmission line. The results were successfully compared against the results in the literature and against a traditional stability simulation tool. The comparison shows the advantages of using more detailed modelling, especially when control and protection devices play a major role in the system's response.

6.2 Recommendation for Future Research

This thesis addresses a broad range of issues associated with the modelling and simulation of power system networks. Many aspects of this work can be further investigated and developed, including the following:

1. **Diakoptics versus Sparsity?** This question has been posed since the 1960s, and some of the reviewers of our recent paper have reinstated it. Sparsity techniques have not been used in this thesis, but we noted that the efficiency of the multilevel partitioning approach depends on the sparse nature of the network. The actual comparison of the two approaches and the possibility of employing sparsity within the MATE concept is still to be investigated.

2. **Optimal network partitioning algorithm.** Optimal partitioning that takes into account the multilevel approach needs to be developed. The algorithm has to take into account the particular system's topology and characteristics.
3. **Fixed point versus Newton-type iteration method?** The iterative procedure for nonlinearities implemented in this work corresponds to the fixed point iteration method. Newton's method requires more processing time due to the calculation of the Jacobian matrix, but its convergence is faster. The possibility of implementing Newton's method for the iteration of nonlinear equations within the Multilevel MATE concept needs to be further investigated.

4. **Expansion of the DFIG wind turbine system model.** The wind turbine system implemented and tested in OVNI does not include models of the protection devices, for example crowbar protection. Blade pitch control and over and under speed trips should also be included.

5. **Stability studies of integration of a DFIG wind turbine farm into the power system network.** The wind turbine model described in this thesis could be used for stability analysis of large wind farms interconnected with the traditional power system network. It could also be used for benchmarking the approximate wind-farm models used in transient stability tools.

6. **Implementation of latency and eigenvalue analysis into OVNI.** Methods have already been developed in our Power System Group to take advantage of the slow and fast nature of the subsystems as reflected in the size of the time step used for their integration, eventually resulting in even more efficient simulation speeds. Eigenvalue analysis provides a valuable insight for stability considerations of power system networks. These approaches need to be implemented into the current version of OVNI.

7. **Development and implementation of power electronic models into OVNI.** This thesis uses an average model of a voltage-source inverter, which is considered appropriate for stability analysis considerations. Detailed models, however, need to be developed and implemented in OVNI that take advantage of the multilevel partitioning approach. The availability of detailed models is necessary for specific types of studies requiring highly detailed modelling or to validate approximate models and their application limitations.

8. **Coupling of OVNI-NET hardware solution to the new version of OVNI and benchmarking of a large power system network model.** Fast simulation of large power system networks is attainable through a combination of software and hardware solutions. The software solution for OVNI's implementation with Multilevel MATE needs to be coupled with the hardware solution of OVNI-NET and tested on an extensive example of a large power system network.
Bibliography


Appendices
Appendix A

Trapezoidal Integration Rule

The application of the implicit trapezoidal integration rule in solving electrical networks was first introduced by Dommel in [4]. To demonstrate the use of the trapezoidal rule, the basic relationship of voltage and current for an inductance is considered:

\[ v_L(t) = L \frac{di_L(t)}{dt} \]  \hspace{1cm} (A.1)

Rewriting the equation and integrating both sides, we obtain

\[ \int_{t_k-\Delta t}^{t_k} v_L(t) \, dt = L \int_{t_k-\Delta t}^{t_k} di_L(t) \]  \hspace{1cm} (A.2)

where \( t_k \) is a discrete time point and \( \Delta t \) is a discretization time step. The integral on the right-hand side is trivial, but to calculate the integral on the left side we need to know the function \( v_L(t) \) between the points \( v_L(t_k) \) and \( v_L(t_k - \Delta t) \). The trapezoidal rule assumes that the function is linear between two consecutive discretization steps (Fig. A.1), therefore the integral can be calculated as

\[ \int_{t_k-\Delta t}^{t_k} v_L(t) \, dt = \text{area} = \frac{v_L(t_k) + v_L(t_k - \Delta t)}{2} \cdot \Delta t \]  \hspace{1cm} (A.3)

A similar approach is applied for the capacitance to obtain its discrete-time equivalent.

129
Figure A.1: Illustration of integration with the trapezoidal rule
Appendix B
Reference Frame Transformations

The introduction of reference frame theory in the analysis of electrical machines has proved useful not only for analysis but has also provided a powerful tool for the implementation of sophisticated control techniques. We present an overview of the most commonly used reference frame transformations. For control purposes, only symmetrical operation is considered.

B.1 Clarke Transformation

Three-phase AC machines are conventionally modelled using phase-variable notation. However, for a three-phase, star-connected machine, the phase quantities are not independent variables but, for symmetrical operation,

\[ i_a(t) + i_b(t) + i_c(t) = 0 \]
\[ v_a(t) + v_b(t) + v_c(t) = 0 \]
\[ \psi_a(t) + \psi_b(t) + \psi_c(t) = 0 \] (B.1)

where the variables denote stator phase currents, voltages and flux linkages, respectively.

As a result of this redundancy, it is possible to transform the phase-variable representation into an equivalent two-phase representation. The transformation from three-phase to two-phase quantities proposed by Clarke [48], is written in matrix form as

\[
\begin{bmatrix}
  i_{sa}(t) \\
  i_{sb}(t)
\end{bmatrix}
= \frac{2}{3}
\begin{bmatrix}
  1 & \cos\left(\frac{2\pi}{3}\right) & \cos\left(\frac{4\pi}{3}\right) \\
  0 & \sin\left(\frac{2\pi}{3}\right) & \sin\left(\frac{4\pi}{3}\right)
\end{bmatrix}
\begin{bmatrix}
  i_a(t) \\
  i_b(t) \\
  i_c(t)
\end{bmatrix}
\] (B.2)

The transformation is equally valid for the voltages and flux linkages. The stator current space vector is defined as the complex quantity

\[ I_s(t) = i_{sa}(t) + j i_{sb}(t) \] (B.3)
Equation (B.2) can be written in a more compact form by introducing a vector rotation factor, \( a = e^{j \frac{2\pi}{3}} \), as follows:

\[
\mathbf{I}_s(t) = \frac{2}{3} \left[ i_a(t) + ai_b(t) + a^2 i_c(t) \right]
\]  

(B.4)

The choice of the constant in the transformations (B.2), (B.4) is somewhat arbitrary. The value of 2/3 has the advantage that magnitudes are preserved across the transformation. A sinusoidal phase current with a peak magnitude of \( I_m \) will produce a current space vector with a peak magnitude of \( I_m \). Since any digital current control scheme is designed to control current amplitude, this form has proved to be more suitable for control purposes. For the Clarke transformation, the inverse relationship can be written as

\[
\begin{bmatrix}
  i_a(t) \\
  i_b(t) \\
  i_c(t)
\end{bmatrix} = \frac{2}{3} \begin{bmatrix}
  1 & 0 & 1 \\
  \cos \left( \frac{2\pi}{3} \right) & \sin \left( \frac{2\pi}{3} \right) & 0 \\
  \cos \left( \frac{4\pi}{3} \right) & \sin \left( \frac{4\pi}{3} \right) & 0
\end{bmatrix} \begin{bmatrix}
  i_{sa}(t) \\
  i_{sb}(t) \\
  i_{sc}(t)
\end{bmatrix}
\]  

(B.5)

### B.2 \( dq \) Transformation

The stator current, voltage and flux-linkage space vectors are complex quantities defined in a reference frame whose real axis is fixed to the magnetic axis of stator winding \( a \). The corresponding quantities defined for the rotor circuit of a three-phase AC machine are similarly defined in a reference frame fixed to the rotor. In the analysis of electrical machines, it is generally necessary to adopt a common reference frame for both rotor and stator. For this reason, a second transformation, known as \( dq \) transformation, is formulated that rotates space-vector quantities by a known angle \( \theta \). While in the Clarke transformation the axes of the space-vector plane are stationary, here a new reference frame is defined where the axes are made to rotate at the same rate as the angular frequency of the phase quantities. This results in stationary current, voltage and flux-linkage space vectors.

The current space vector in the \( dq \) rotating reference frame can be written as

\[
\mathbf{I}_{dq} = i_{sd} + j i_{sq} = \mathbf{I}_s e^{-j \theta}
\]  

(B.6)

which can be written in matrix form as

\[
\begin{bmatrix}
  i_{sd} \\
  i_{sq}
\end{bmatrix} = \begin{bmatrix}
  \cos(\theta) & \sin(\theta) \\
  -\sin(\theta) & \cos(\theta)
\end{bmatrix} \begin{bmatrix}
  i_{sa} \\
  i_{sb}
\end{bmatrix}
\]  

(B.7)

The real component of the current space vector in this new reference frame is
called the direct (d-) axis component, while the imaginary component is called the quadrature (q-) axis component. The main advantage of the dq transformation is the elimination of position dependency from the machine electrical variables.

The inverse dq transformation can be written in matrix form as

$$\begin{bmatrix} i_{sa} \\ i_{sb} \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix}$$  \hspace{1cm} (B.8)$$

The dq transformation performed directly on the phase quantities has the following well-known form:

$$\begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} = \frac{2}{3} \begin{bmatrix} \cos(\theta) & \cos(\theta - \frac{2\pi}{3}) & \cos(\theta - \frac{4\pi}{3}) \\ -\sin(\theta) & -\sin(\theta - \frac{2\pi}{3}) & -\sin(\theta - \frac{4\pi}{3}) \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$  \hspace{1cm} (B.9)$$
Appendix C

Digital Pass-Band Filter for Filtering Stator-Flux

To accurately calculate the stator-flux vector position $\theta_{sf}$ in Section 5.6.3, it is necessary to filter out DC offsets in the stator flux. Instantaneous values of stator flux linkages $\psi_a$, $\psi_b$ and $\psi_c$ are filtered using the digital pass-band filter described here. A simple RLC parallel circuit in resonance may be used to eliminate the DC component of the stator-flux linkages. Fig. C.1 depicts the circuit configuration for parallel resonance.

![Circuit configuration for filtering 60 Hz component of $\psi_a(t)$](image)

Figure C.1: Circuit configuration for filtering 60 Hz component of $\psi_a(t)$

The admittance of this circuit can be expressed as

$$Y = \frac{1}{R_F} + j \left( \omega C_F - \frac{1}{\omega L_F} \right) \quad (C.1)$$

In the resonant condition, the susceptance is equal to zero, which leads to the following well-known expression for the resonant frequency:

$$\omega = \omega_0 = \frac{1}{\sqrt{L_F C_F}} \quad (C.2)$$

The resonant frequency is entirely specified by the choice of $L_F$ and $C_F$. The magnitude of the filtered flux linkage $\Psi_{a,F}$ will vary as a function of the frequency of the input flux linkage $\Psi_a$, as depicted in Fig. C.2.
Appendix C. Digital Pass-Band Filter for Filtering Stator-Flux

![Diagram: Variation of filtered flux-linkage magnitude as a function of frequency]

Figure C.2: Variation of filtered flux-linkage magnitude as a function of frequency

The bandwidth of the resonant curve is noted as $\omega_{BW}$. By forming the ratio of the resonant frequency to the bandwidth, we obtain a factor that is a measure of the selectivity or sharpness of tuning of the parallel RLC circuit. For a parallel resonant circuit, the quality factor is

$$Q_0 = \frac{\omega_0}{\omega_{BW}} = \omega_0 R_F C_F$$  \hspace{1cm} (C.3)

With the resonant frequency set to 50 Hz (314 rad/s) for the particular example in Chapter 5 and with the desired quality factor, the values of $R_F$, $L_F$ and $C_F$ can be determined from the above equations by freely choosing either one of them.

The circuit in Fig. C.1 can be discretized using the trapezoidal rule of integration. The equivalent discrete circuit is depicted in Fig. C.3, where

$$h_{LF}(t) = -\frac{\Delta t}{L_F} \psi_a(t - \Delta t) + h_{LF}(t - \Delta t)$$
$$h_{CF}(t) = \frac{4C_F}{\Delta t} \psi_a(t - \Delta t) - h_{CF}(t - \Delta t)$$  \hspace{1cm} (C.4)

By solving the circuit, the filtered stator-flux linkage of phase $a$ is

$$\psi_a(t) = \frac{\psi_a(t) + h_{LF}(t) + h_{CF}(t)}{\frac{1}{R_F} + \frac{\Delta t}{2L_F} + \frac{2C_F}{\Delta t}}$$  \hspace{1cm} (C.5)

135
Appendix C. Digital Pass-Band Filter for Filtering Stator-Flux

Figure C.3: Equivalent circuit of the parallel RLC filter discretized with the trapezoidal rule

\[
\frac{\psi(t)}{R_F} \quad \quad R_F \quad \frac{2L_F}{\Delta t} \quad h_L \quad \frac{\Delta t}{2C_F} \quad h_C \quad \psi_{e,f}(t)
\]
Appendix D

Test System Parameters

D.1 Parameters of the 2250 hp Induction Motor from Chapter 4.4.1

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rating</td>
<td>2250 hp</td>
</tr>
<tr>
<td>Line-to-line voltage</td>
<td>2300 V</td>
</tr>
<tr>
<td>Poles</td>
<td>4</td>
</tr>
<tr>
<td>Speed</td>
<td>1786 rpm</td>
</tr>
<tr>
<td>Base torque</td>
<td>$8.9 \cdot 10^3 , Nm$</td>
</tr>
<tr>
<td>Rated current</td>
<td>421.2 A</td>
</tr>
<tr>
<td>Inertia</td>
<td>$63.87 , kg , m^2$</td>
</tr>
<tr>
<td>$R_s$</td>
<td>0.029 $\Omega$</td>
</tr>
<tr>
<td>$R'_r$</td>
<td>0.022 $\Omega$</td>
</tr>
<tr>
<td>$X_{ls}$</td>
<td>0.226 $\Omega$</td>
</tr>
<tr>
<td>$X'_{lr}$</td>
<td>0.226 $\Omega$</td>
</tr>
<tr>
<td>$X_m$</td>
<td>13.04 $\Omega$</td>
</tr>
</tbody>
</table>

Table D.1: 2250 hp induction motor parameters
D.2 Parameters of the 835 MVA Synchronous Generator from Chapter 4.4.2

Rating: 835 MVA
Line-to-line voltage: 26 kV
Poles: 2
Speed: 3600 rpm
Combined inertia of generator and turbine: $J = 0.0658 \cdot 10^6 \text{ J s}^2$

$R_s = 0.00243 \Omega$
$X_{ls} = 0.1538 \Omega$

$X_q = 1.457 \Omega$  $X_d = 1.457 \Omega$

$R_{kq1}' = 0.00144 \Omega$  $R_{fd}' = 0.00075 \Omega$

$X_{lkq1}' = 0.6578 \Omega$  $X_{lf'd}' = 0.1145 \Omega$

$R_{kq2}' = 0.00681 \Omega$  $R_{kd}' = 0.01080 \Omega$

$X_{lkq2}' = 0.07602 \Omega$  $X_{lkd}' = 0.06577 \Omega$

Table D.2: 835 MVA steam turbine generator parameters
### D.3 Parameters of the 7.5 kW Doubly Fed Induction Generator from Chapter 5.10

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated power</td>
<td>7.5 kW</td>
</tr>
<tr>
<td>Stator voltage</td>
<td>415 V</td>
</tr>
<tr>
<td>Rotor voltage</td>
<td>440 V</td>
</tr>
<tr>
<td>Rated stator current</td>
<td>19 A</td>
</tr>
<tr>
<td>Rated rotor current</td>
<td>11 A</td>
</tr>
<tr>
<td>Pole pairs</td>
<td>3</td>
</tr>
<tr>
<td>Rated speed</td>
<td>970 rmp</td>
</tr>
<tr>
<td>Base frequency</td>
<td>50 Hz</td>
</tr>
<tr>
<td>(N_s/N_r)</td>
<td>1.7</td>
</tr>
<tr>
<td>(J)</td>
<td>7.5 kgm(^2)</td>
</tr>
<tr>
<td>Stator connection</td>
<td>delta</td>
</tr>
<tr>
<td>Rotor connection</td>
<td>wye</td>
</tr>
</tbody>
</table>

Machine resistances and inductances per phase:
- \(R_s = 1.06 \, \Omega\)
- \(R_r = 0.80 \, \Omega\)
- \(L_s = 0.0664 \, \text{H}\)
- \(L_o = 0.0810 \, \text{H}\)
- \(L_r = 0.0320 \, \text{H}\)

Control parameters:
<table>
<thead>
<tr>
<th></th>
<th>Stator-side converter</th>
<th>Rotor-side converter</th>
</tr>
</thead>
<tbody>
<tr>
<td>(K_v)</td>
<td>0.12</td>
<td>0.49</td>
</tr>
<tr>
<td>(a_v)</td>
<td>0.9248</td>
<td>0.988</td>
</tr>
<tr>
<td>(K_i)</td>
<td>4.72</td>
<td>(K_{ir}) = 20</td>
</tr>
<tr>
<td>(a_i)</td>
<td>0.96</td>
<td>(a_{ir}) = 0.985</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>Rotor-side converter</th>
</tr>
</thead>
<tbody>
<tr>
<td>(K_{ir})</td>
<td>20</td>
</tr>
<tr>
<td>(a_{ir})</td>
<td>0.985</td>
</tr>
</tbody>
</table>

Table D.3: 7.5 kW doubly fed induction generator wind turbine system parameters