METHODS FOR SYNTHESIS OF MULTIPLE-INPUT TRANSLINEAR ELEMENT NETWORKS

A Dissertation
Presented to
The Academic Faculty

By

Shyam Subramanian

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
in
Electrical and Computer Engineering

School of Electrical and Computer Engineering
Georgia Institute of Technology
December 2007

METHODS FOR SYNTHESIS OF MULTIPLE-INPUT TRANSLINEAR ELEMENT NETWORKS

Approved by:

Dr. David V. Anderson, Advisor
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. Paul Hasler
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. James H. McClellan
Associate Professor, School of ECE
Georgia Institute of Technology

Dr. Thomas G. Habetler
Professor, School of ECE
Georgia Institute of Technology

Dr. Bradley A. Minch
Associate Professor of Electrical and Computer Engineering
Franklin W. Olin College of Engineering

Date Approved: July 16, 2007

Dedicated to my parents and Kishan - for everything!

ACKNOWLEDGMENTS

श्री गणेशाय नमः

At the outset, I would like to thank my advisor Dr. David Anderson for his constant support and encouragement and for giving me the freedom to explore different approaches to solving the problems under consideration. I would like to thank Dr. Brad Minch for his many suggestions and corrections. Dr. Minch is the father of MITE networks, so the influence of his work on mine is gratefully acknowledged. I would like to thank Dr. Hasler for reviewing my work from time to time and Dr. McClellan and Dr. Habetler for their comments about the work. I would like to offer my heartfelt thanks to Dr. Rajbabu Velmurugan for the joint work involving the application of MITE networks to particle filters and for taking an active interest in my research. I am grateful to Kofi Odame for the numerous discussions and for being patient enough to listen even when I was boring him with details of my research. This research was partially supported by Texas Instruments and hence, I would also to like to thank my TI mentor Fernando Mujica for reviewing my work from time to time.

Both from a personal and technical point of view, the members of the CADSP lab have been excellent company for the past five years and I sincerely thank them for that. I am grateful to my friend Badri Narayanan V.R. for his company and also for the considerable amount of negative feedback, freely given by him, which helped me in many ways. Ganesh, Arumugam, and Kannan have been excellent roommates and friends and I would like to thank them. Dr. M. A. Reddy, formerly of the Indian Institute of Technology Madras, is the single person responsible for my interest in circuits and has influenced me a lot through the sheer force of his personality and his views. Whatever little I know in analog circuits is the result of the wonderful magic woven by him in his classes - if I have not learnt more, it is the fault of my laziness rather than any flaw in his teaching. My sincerest thanks to him.

Lastly, the foremost of my thanks go to my family who have been a constant source of encouragement and support without which I would not have been able to finish this work. I would, therefore, like to dedicate this thesis to my parents and brother.

TABLE OF CONTENTS

ACKNOWLEDGMENTS iv

LIST OF TABLES viii

LIST OF FIGURES ix

SUMMARY xv

CHAPTER 1 INTRODUCTION AND BACKGROUND 1
- 1.1 Translinear Circuits 1
- 1.2 The Multiple-Input Translinear Element 2
  - 1.2.1 MITE implementations 3
- 1.3 Product–of–power–law Networks 3
  - 1.3.1 Stability of the Operating Point of POPL Networks 8
  - 1.3.2 Sensitivity Considerations 9
- 1.4 Previous synthesis methods 11
  - 1.4.1 Synthesis of Static MITE Networks 11
  - 1.4.2 Synthesis of Dynamic MITE Networks 15

CHAPTER 2 CONDITIONS ON MATRICES ASSOCIATED WITH MITE CIRCUITS 21
- 2.1 Static Modeling of the Nonideal MITE 21
- 2.2 Mathematical Preliminaries 23
- 2.3 POPL Networks 25
- 2.4 The General MITE Network 32
- 2.5 Robust Criteria for Uniqueness of the Operating Point 34
- 2.6 Stability of the Operating Point of POPL Networks 38
  - 2.6.1 New stability criterion for POPL networks 39
- 2.7 Appendix 2.A 42

CHAPTER 3 SYNTHESIS OF MITE TRANSLINEAR LOOPS 44
- 3.1 Translinear Loops 44
- 3.2 Reformulation of POPL Networks 46
- 3.3 The Synthesis Problem 46
- 3.4 Operating Point Uniqueness and Stability 49
- 3.5 Solution Methodology 49
- 3.6 Simple Synthesis Procedure 53
- 3.7 Linear Diophantine Equations 57
- 3.8 Existence and Construction of Solution 58
- 3.9 Optimal Synthesis Algorithm 60
- 3.10 Example 60
- 3.11 Appendix 3.A 61
  - 3.11.1 MATLAB code for the simple synthesis procedure in Section 3.6 . . 61
  - 3.11.2 MATLAB code for finding the solution(s) Z given translinear loop matrix A 63
  - 3.11.3 MATLAB code for finding the hilbert basis of A 64

3.11.4 MATLAB code for forming solution matrices Z from hilbert basis . 66
3.12 Conclusion 70

CHAPTER 4 SYNTHESIS OF 2-MITE POPL NETWORKS 71
4.1 Mathematical Preliminaries 72
4.2 Uniqueness and stability of Operating Point 72
4.3 2-MITE network graphs 74
4.4 Necessary conditions 79
4.5 Single-Output POPL Networks 86
4.5.1 Necessary and sufficient conditions for Λ to be 2-MITEable 86
4.5.2 Case when Λ has no powers that are $\pm 1/2$ 87
4.5.3 Case when Λ has some powers that are $\pm 1/2$ 93
4.6 2-MITE synthesis of arbitrary POPL equations with a single output 93
4.6.1 General case: Rational power matrix 102
4.7 2-MITE synthesis for partially reconfigurable POPL networks : The MITE FPAA 103
4.8 Coates graph analysis of general POPL networks 109
4.9 2-MITE POPL networks with two outputs 114
4.10 Appendix 4.A 119
4.11 Conclusion 127

CHAPTER 5 SYNTHESIS OF MITE LOG-DOMAIN FILTERS WITH UNIQUE OPERATING POINTS 128
5.1 Introduction 128
5.2 Mathematical Preliminaries 129
5.3 Constraints on the State-Space Equations 129
5.4 Synthesis Procedure 132
5.5 Uniqueness of the Operating Point 134
5.6 Conclusion 135

CHAPTER 6 SYNTHESIS EXAMPLES 140
6.1 Synthesis of static functions 140
6.1.1 Current Splitters 140
6.1.2 Particle filters and target tracking 141
6.1.3 Implementing static functions with a geometric current splitter . . 144
6.1.4 Implementation of the inverse tangent function 144
6.1.5 Implementation of the Gaussian 148
6.2 Synthesis of dynamical systems 152
6.2.1 The Lorenz system 153
6.2.2 A sinusoidal oscillator with independent frequency and amplitude control 155
6.2.3 Chip fabrication 162

CHAPTER 7 CONCLUSIONS AND FUTURE RESEARCH 167
7.1 Contributions of this research 167
7.2 Future Research 169
7.2.1 Future Theoretical research 169
7.2.2 Future Practical Research 170

CHAPTER 8 NOTATION 171

REFERENCES 173

LIST OF TABLES

Table 4.1 The functional synthesizability of 2-MITE networks for 3 and 4 inputs 105

Table 6.1 Simulated characteristics of the arctan circuit 148

Table 6.2 Simulation results of the gaussian circuit 152

LIST OF FIGURES

Figure 1.1 Symbol for a n-input multiple-input translinear element (MITE). . . . 2

Figure 1.2 Schemes for constructing a n-input MITE. In the scheme in (a), the BJT (MOSFET) is in the common-emitter (common-source) configuration while in the scheme in (d), it is in a common-base (common-gate) configuration. Implementations of (a) are shown in (b) and (c), the former with a passive summer and the latter with an active summer. An implementation of (d) is shown in (e). 4

Figure 1.3 Cascoded floating-gate implementations of MITEs. The PFET implementation in (a) is the practical one; the NFET implementation in (b) is often used for illustrative purposes. 5

Figure 1.4 The general form of the MITE network implementing a POPL function. The output currents are a product of the input currents raised to different powers. 5

Figure 1.5 Stability analysis of the POPL network. Capacitances are attached from each node V_i to ground. The currents through the gates of all MITEs are neglected. 8

Figure 1.6 The cascade network implementing the equation $\prod_{k \text{ odd}}^{n'} I'_k = \prod_{k \text{ even}}^{n'} I'_k$. 15

Figure 1.7 The MITE implementation of the i^{th} equation in the set of equations $\dot{\mathbf{I}}_x = f(\mathbf{I}_x, \mathbf{I}_u)$. The state variable I_{xi} is transformed into V_i through $I_{xi} = \alpha_i \exp(\beta_i V_i)$. The noninverting output structure for $\beta_i > 0$ is shown in (a) and the inverting output structure for $\beta_i < 0$ is shown in (b). The static MITE networks take as inputs the vector variables $\mathbf{I}_x$ and $\mathbf{V}$ and produces outputs $(C_i/\beta_i)f_i^+/I_{xi}$ and $(C_i/\beta_i)f_i^-/I_{xi}$. 18

Figure 1.8 The standard MITE first–order lowpass filter. The filter obeys the equation $(CU_{\mathrm{T}})/\kappa \dot{I}_y + I_{\tau 1} I_y = I_{\tau 2} I_x$. 20

Figure 2.1 (a) Symbol for a n-input MITE. (b) Symbol of a PFET modeled by Equation (2.11). The same symbol is used for a cascoded PFET. . . . 23

Figure 2.2 Example comparing the uniqueness criterion in Theorem 2.3.3 and the ideal uniqueness criterion. (a) Two circuits implementing $I_{\mathrm{o}1} = I_{\mathrm{i}1}^{-1} I_{\mathrm{i}3}^2$; $I_{\mathrm{o}2} = I_{\mathrm{i}2}^{-1} I_{\mathrm{i}3}^2$. (b) Circuits for finding the open-loop transfer characteristics. (c) Plots of the open-loop circuits for $I_{\mathrm{i}2}$ varied logarithmically from $50nA$ to $500nA$. 30

Figure 2.3 The dc circuit corresponding to a general MITE network described in Section 2.4. 31

Figure 2.4 The small-signal equivalent model used for analyzing the stability properties of a POPL MITE network. v_{g} is the small-signal floating-gate voltage. C is the unit capacitance that the floating-gate capacitances are made of. C_{p} is the sum of the capacitances from the floating-gate to bulk, source, and drain. As a first-order approximation, the other capacitances in the network are neglected. 40

Figure 3.1 The canonical MITE network used to implement STLE Equation (3.1). The voltages $V_1, V_2, \ldots, V_n$ are generated by "diode" connecting them to the respective drains of the input MITEs with currents $I_1, I_2, \ldots, I_n$. 46

Figure 3.2 Synthesis of the MITE network implementing $I_4 = I_1 I_2^{1/2} I_3^{-1/2}$; $I_5 = I_1^2 I_2 I_3^{-2}$ described in steps $1-4$. The MITE network in (a) is obtained by assuming the input connectivity matrix to be the identity and the output connectivity matrix to be Λ. The weights in (a) are rendered nonnegative by adding a weight 2 to all weights connected to V_3, which results in the network in (b). The nonnegative weights in (b) are converted into nonnegative integers by multiplying all weights connected to V_2 and V_3 by 2, which results in (c). The final network in (d) is obtained by using Theorem 3.5.2 i.e., the completion theorem. 55

Figure 4.1 The general form of the MITE network implementing a POPL function. The output currents are a product of the input currents raised to different powers. 71

Figure 4.2 (a) A single-output 2-MITE network. (b) The Coates graph $G_{\mathrm{c}}(X)$ of the input-connectivity matrix X of the network (c) The Coates graph $G_{\mathrm{c}}(\widehat{X})$ of the reduced input-connectivity matrix $\widehat{X}$. 76

Figure 4.3 (a) A component of the Coates graph $G_{\mathrm{c}}(X)$ of the input-connectivity matrix X of a 2-MITE POPL network with directed circuit C. (b) The Coates graph $G_{\mathrm{c}}(\widehat{X})$ of the reduced input-connectivity matrix of the same network 77

Figure 4.4 (a) The circuit C' used in the proof of 2 in Theorem 4.3.1. It is a hypothetical undirected circuit assumed to exist in the graph underlying G. The directed edge (i_1, i_2) is assumed to exist in G. (b) The proof shows that if we assume (i_{m-1}, i_m) to be a directed edge in G, then (i_m, i_{m+1}) has to be the directed edge connecting i_m and i_{m+1} in G. . . 79

Figure 4.5 Calculation of $\det(X)$ using Equation (4.4). The edges in the 1-factors in $G_{\mathrm{c}}(X)$ are shown with continuous edges while the edges not belonging to the 1-factor are shown by dotted edges. (a) The 1-factor formed by all self-loops in $G_{\mathrm{c}}(X)$. (b) The 1-factor formed by C and the self-loops attached to vertices not in C. 81

Figure 4.6 Calculation of Δ_{ii} using Equation (4.5). The edges in the 1-factors in $G_c(X) \setminus i$ are shown with continuous edges while the edges not belonging to the 1-factor are shown by dotted edges. (a) When $i \in C$, all self-loops in $G_c(X) \setminus i$ form the only 1-factor. (b) When $i \notin C$, all self-loops in $G_c(X) \setminus i$ form one of the two 1-factors. (c) When $i \notin C$, the second 1-factor is formed by C and the self-loops in $G_c(X) \setminus i$ attached to vertices not in C. 82

Figure 4.7 Calculation of Δ_{ji} using Equation (4.5). The edges in the 1-factorials involved are shown with continuous edges while the edges not belonging to the 1-factorials are shown by dotted edges. (a) When $j \in C$, P_{ji} and all self-loops in $G_c(X) \setminus P_{ji}$ form the 1-factorial. (b) When $j \notin C$ and there is no path from j to i, then no 1-factorial exists. (c) When $j \notin C$ and there is a P_{ji}, a 1-factorial is formed by P_{ji}, and the self-loops in $G_c(X) \setminus P_{ji}$. (d) When $j \notin C$ and there is a P_{ji}, another 1-factorial is formed by P_{ji}, C and the self-loops in $G_c(X) \setminus C \cup P_{ji}$. 84

Figure 4.8 $G_c(\widehat{X})$ for 2-MITE POPL networks with single outputs that has no $\pm 1/2$ powers and satisfies $v \neq \delta$. (a) is the graph $\widetilde{G}$ formed when the directed circuit in $G_c(\widehat{X})$ is contracted to a vertex v that is not equal to δ. (b) is the $G_c(\widehat{X})$ corresponding to the same $\widetilde{G}$. (c) is the case when the sum of ± 2-powers in Λ is $+2$. (d) is followed when the sum of ± 2-powers is -2. (e) is found when the sum of ± 2-powers is 0 and when $\Lambda_\delta = +2$. (f) is generated when the ± 2-powers add up to 0 and when $\Lambda_\delta = -2$. The cases (c)-(f) occur only $d(\beta, \delta) - d(\gamma, \delta)$ is even. In the odd case the powers occur as shown in (g). The sequence of Λ_j values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path. 88

Figure 4.9 $G_c(\widehat{X})$ for 2-MITE POPL networks with single outputs that has no $\pm 1/2$ powers and satisfies $v = \delta$. (a) is the graph $\widetilde{G}$ formed when the directed circuit in $G_c(\widehat{X})$ is contracted to a vertex v. (b) is the $G_c(\widehat{X})$ corresponding to the same $\widetilde{G}$. Without loss of generality, $\Lambda_\epsilon \neq 0$ has been assumed. The sequence of Λ_j values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path. 92

Figure 4.10 $G_c(\widehat{X})$ for 2-MITE POPL networks with single outputs that has some $\pm 1/2$ powers. (a) is the general form of $G_c(\widehat{X})$ required when $\pm 1/2$ powers are present. (b) is the case when the sum of $\pm 1/2$-powers in Λ is $+1$. (c) is followed when the sum of $\pm 1/2$-powers is -1. (d) is found when the sum of $\pm 1/2$-powers is 0. The sequence of Λ_j values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path. 94

Figure 4.11 Optimal 2-MITE synthesis of the equation $I_3 = I_1^{1/4} I_2^{3/4}$. The only minimal solution for this equation is got by implementing $I_1 I_2 I_2^2 I_3^{-2} I_3^{-2} = 1$ or, equivalently, as $I_3 = I_1^{1/2} I_2^{1/2} I_2^1 I_3^{-1}$. Clearly, the presence of the 1/2 means that 2 components are needed for the synthesis. Two slightly different networks result. The Coates graph $G_c(X)$ of the reduced input-connectivity matrices of these solutions along with the relevant powers is shown in (a) and (c). The corresponding complete MITE networks are shown in (b) and (d). The presence of the current mirrors makes the general case of rational powers different from the case where the powers are integral. 103

Figure 4.12 The scheme used to maximize the number of synthesizable functions while avoiding fine granularity in the MITE FPAA. The input section of the POPL network is fixed. Multiple functions are obtained by changing the connectivity of the output MITE. This reconfiguration is done through a switch matrix. 105

Figure 4.13 (a), (b), and (c) are the only connected 2-MITE POPL networks with 3 inputs i.e., $n = 3$. From Table 4.1, it follows that (a) can synthesize the maximum number of functions for $n = 3$, while (b) and (c) fall short : (b) and (c) cannot synthesize $I_o = I_1^2 I_2^{-2} I_3$ and (c) can implement $I_o = I_1^2 I_2^{-1}$ only by using copies of currents in $I_1 I_2 I_3^{-1}$. 106

Figure 4.14 (a), (b), (c), (d), and (e) are the only connected 2-MITE POPL networks with 4 inputs i.e., $n = 4$. From Table 4.1, it follows that (a) can synthesize the maximum number of functions for $n = 4$, while everything else falls short : (b) and (c) cannot synthesize $I_o = I_1^2 I_2^{-2} I_3^2 I_4^{-1}$ and $I_o = I_1^2 I_2^{-1} I_3 I_4^{-1}$ and (c), in addition, cannot implement $I_o = I_1 I_2^{-2} I_3 I_4$ too. (d) and (e) have obviously very limited functional synthesizability. 107

Figure 4.15 The basic structure used to implement almost any 2-MITEable single-output POPL function with at most n inputs. 107

Figure 4.16 (a) The vertex j and its successors $i_1, i_2, \ldots, i_s$ in the Coates graph of a general POPL MITE network. The power of j, Λ_j, is related to the powers of $i_1, i_2, \ldots i_s$ and to the source at j through Equation (4.35). (b) The graph of (a) for the particular case when the fan–in is 2. In this case, every vertex j has a unique parent $\alpha(j)$. Barring the degenerate case of C being a loop, the weights of all edges is now unity 110

Figure 4.17 Transformation of a MCGR of a POPL network into the Coates graph of the network. 112

Figure 4.18 Construction of a MCGR of a 2-MITE POPL network satisfying Theorem 4.8.1. 115

Figure 4.19 The modified Coates graphs of the only structurally distinct Coates graphs of X for the single-output case. The length of each edge is also mentioned. Note that the effect of the powers of the successors of j can be calculated from Equation (4.37) 116

Figure 4.20 The only possible connected distinct graphs representing the MCGRs of 2-MITE POPL networks with two outputs. The elements of the power matrices are all in $\{0, \pm 2, \pm 1\}$: no $\pm 1/2$ powers exist. 117

Figure 4.21 The only possible disconnected MCGRs of 2-MITE POPL networks with two outputs. The power matrices necessarily have fractional ($\pm 1/2$) powers, except in (a),(b), and (c) under some conditions. 118

Figure 4.22 Different powers generated by the MCGRs in Figure 4.20(a) and (c). . 120

Figure 4.23 Different powers generated by the MCGR in Figure 4.20(b). 121

Figure 4.24 Different powers generated by the MCGR in Figure 4.20(d). 122

Figure 4.25 The input-connectivity matrix X can be written, by a permutation similarity, in the form shown in Equation (4.39). $G_{\mathrm{c}}(X_1)$ represents the directed circuit C, $G_{\mathrm{c}}(X_3)$ represents $G_{\mathrm{c}}(X) - C$, $G_{\mathrm{c}}(X_2)$ represents the edges connecting $G_{\mathrm{c}}(X_1)$ and $G_{\mathrm{c}}(X_3)$. 124

Figure 5.1 The circuit blocks used in implementing first-order equations 137

Figure 5.2 A generic shifted-companion-form filter. It can be used to generate any linear transfer function. The multiplier block is implemented as described in Section 5.4. 138

Figure 5.3 Symbol for a 2-MITE. Ideally, it should obey the law $I = I_{\mathrm{s}} \exp(\frac{\kappa}{U_{\mathrm{T}}}(V_1 + V_2))$, where I_{s} is a scaling current. 139

Figure 6.1 Two MITE circuits for the geometric current splitter. The current I_x is split into two currents I_p and I_n such that their geometric mean is a fixed bias current I_b. 142

Figure 6.2 The version of the circuit in Figure 6.1(b) with PFET floating-gate MOSFETs for the MITEs. 143

Figure 6.3 Simulation of the currents in a MITE geometric current splitter for the circuit shown in Figure 6.2.The value of I_b is 10nA. 143

Figure 6.4 The MITE circuit implementing the arctan function. Here, $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a - 1.12 I_a$ and $I_y = (I_{x+}I_a - I_{x-}I_a)/(\sqrt{I_a I_r} + 0.63 I_a)$ 145

Figure 6.5 Results of simulation of the Arctan circuit. The currents I_a and I_b are each 10nA. The "function" refers to $\phi(I_x/I_a) = 2/\pi \arctan(I_x/I_a)$, the "approximation" is $f(I_x/I_a) = \frac{I_x}{0.63 I_a + \sqrt{0.88 I_a^2 + I_x^2}}$, and the "Circuit simulation" is the simulation plot of the circuit in Figure 6.4. 147

Figure 6.6 The MITE circuit to implement the Gaussian function. $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a$ 149

Figure 6.7 The results of simulation of the gaussian circuit. The value of I_a is 20nA and the value of I_c is 10nA. 150

Figure 6.8 Simulated behavior of the gaussian circuit. The current I_a determining the standard deviation of the gaussian is varied from 11nA to 31nA in steps of 2nA. 151

Figure 6.9 The standard MITE first-order lowpass filter. The filter obeys the equation $(CU_{\mathrm{T}})/\kappa\dot{I}_y + I_{\tau 1}I_y = I_{\tau 2}I_x$, where $I_{\tau 1}$ need not be constant. . . . 153

Figure 6.10 The bidirectional MITE first-order lowpass filter. The input I_x undergoes splitting and the positive current outputs are fed into firstorder lowpass filters. The filter obeys the equation $(CU_{\mathrm{T}})/\kappa\dot{I}_y + I_{\tau 1}I_y = I_{\tau 2}I_x$, where $I_y = I_{y+} - I_{y-}$. Here, $I_{\tau 1}$ need not be constant. 154

Figure 6.11 The part of the MITE circuit implementing the Lorenz system consisting of the bidirectional lowpass filters. (a) corresponds to the x equation and (b) corresponds to the y equation in Equation (6.11). The input $I_{\mathrm{in},x}$ is also generated in (b). 156

Figure 6.12 The part of the MITE circuit implementing the Lorenz system consisting of the bidirectional lowpass filter corresponding to the z coordinate in Equation (6.11). 157

Figure 6.13 The static MITE circuit implementing the inputs $I_{\mathrm{in},y}$ and $I_{\mathrm{in},y}$, i.e., the right-hand side of the y and z equations in Equation (6.11). 158

Figure 6.14 The results of simulation of the circuit of the Lorenz system for the parameter values $\sigma = 3$, $\beta = 1$, $\rho = 30$. 159

Figure 6.15 The part of the MITE circuit implementing the sinusoidal oscillator consisting of the bidirectional lowpass filters. (a) corresponds to the x equation and (b) corresponds to the y equation in Equation (6.17). Here the time varying current $I_s = (I_r^2 + I_\beta^2)/I_b = (I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2)/I_b$ . . 163

Figure 6.16 The part of the MITE circuit implementing the sinusoidal oscillator consisting of POPL networks implementing $I_{\mathrm{in},x}$, $I_{\mathrm{in},y}$, and $I_s = (I_r^2 + I_\beta^2)/I_b = (I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2)/I_b$ 164

Figure 6.17 The results of circuit simulation of the sinusoidal oscillator for varying amplitudes. The current I_b is varied from 5nA to 25nA in steps of 2nA while $I_a = 8I_b$ and $I_\omega = 20$nA. The ideal behaviour is given in dotted lines for comparison. 165

Figure 6.18 The layout of the chip in 0.5μ technology containing the arctan, gaussian, Lorenz, and the sinusoidal oscillator blocks 166

SUMMARY

Translinear circuits are circuits in which the exponential relationship between the output current and input voltage of a circuit element is exploited to realize various differential or algebraic equations. The precise exponential characteristic of the Bipolar Junction Transistor (BJT) and other devices has been responsible for the increase in popularity of this branch of analog circuits and has stimulated research geared towards implementation of not only linear, but also nonlinear systems in the analog domain. Also, the basic concept of using the exponential characteristic of certain circuit elements has been used in other popular analog components such as bandgap references, Proportional to Absolute Temperature (PTAT) circuits, and constant-gm biasing circuits.

This thesis is concerned with a subclass of translinear circuits, in which the basic translinear element has an output current that is exponentially related to a weighted sum of its input voltages. This multiple-input translinear element (MITE) can be used for the implementation of the same class of functions as traditional translinear circuits. The voltage addition that gives rise to multiplication of currents in traditional translinear circuits is replaced by weighted summation through a capacitive voltage divider. The implementation of algebraic or (algebraic) differential equations using MITEs can be reduced to the implementation of the so–called product–of–power–law (POPL) relationships, in which an output is given by the product of inputs raised to different powers. Hence, the synthesis of these POPL relationships, and their optimization with respect to the relevant cost functions, are very important in the theory of MITE networks.

In this thesis, different constraints on the topology of these POPL networks that result in desirable system behavior are explored and different methods of synthesis, subject to these constraints, are developed. The constraints are usually conditions on certain matrices of the network, which characterize the weights in the relevant MITEs. Some of these constraints are related to the uniqueness of the operating point of the network and the stability of the network. Conditions that satisfy these constraints are developed in this work. The cost functions to be minimized are the number of MITEs and the number of input gates in each MITE. A complete solution to POPL network synthesis is presented here that minimizes

the number of MITEs first and then minimizes the number of input gates to each MITE. A procedure for synthesizing POPL relationships optimally when the number of gates is minimal, i.e., 2, has also been developed here for the single–output case. A MITE structure that produces the maximum number of functions with minimal reconfigurability is developed for use in MITE field–programmable analog arrays. The extension of these constraints to the synthesis of linear filters is also explored, the constraint here being that the filter network should have a unique operating point in the presence of nonidealities. Synthesis examples presented here include nonlinear functions like the inverse tangent function and the gaussian function which find application in analog implementations of particle filters. Synthesis of dynamical systems is presented here using the examples of a Lorenz system and a sinusoidal oscillator. The procedures developed here provide a structured way to automate the synthesis of nonlinear algebraic functions and differential equations using MITEs.

CHAPTER 1
INTRODUCTION AND BACKGROUND

1.1 Translinear Circuits

The term *translinear circuit* was coined by Barrie Gilbert [1] to refer a class of bipolar junction transistor (BJT) circuits whose behavior depended upon the *trans*conductance of a BJT being a *linear* function of its collector current. Such a property is essentially because the collector current is exponentially related to the base-emitter voltage. The modern usage of the term [2] generalizes this to include MOSFETs in the subthreshold region, which exhibit exponential behavior. Roughly speaking, a translinear circuit is one that utilizes the exponential behavior of some of its elements, called *translinear elements*, for the circuit function. Gilbert's translinear principle [1], also known as the *static translinear principle* [3], is the fundamental principle for implementing static or memoryless systems. Mathematically, this is just a circuit version of the well-known fact the logarithm of a product of numbers is the sum of the individual logarithms. A systematic approach to the analysis and synthesis of static translinear circuits was done by Seevinck [4]. Using the translinear principle, numerous static functions like four-quadrant multipliers, two-quadrant dividers, sinusoidal frequency multipliers, vector magnitude operations, trigonometric functions, taylor series expansions have been synthesized [5, 4, 6, 7].

Dynamic translinear circuits are translinear circuits that include capacitors and can be used for the implementation of ordinary differential equations. The *dynamic translinear principle* [3] is essentially an application of the fact that the derivative of an exponential function is proportional to the function itself. Using this property, it is possible to implement systems whose input-output relationship is linear but the circuit operation *internally* is nonlinear. This class of filters, called as log-domain filters or exponential state-space filters, were first introduced by Robert Adams who presented a first-order log-domain filter. A synthesis methodology for these filters was given by Douglas Frey [8,9]. Mulder *et al.* [10,3] established the dynamic translinear principle along with an extension of Seevinck's synthesis procedures to dynamic translinear circuits.

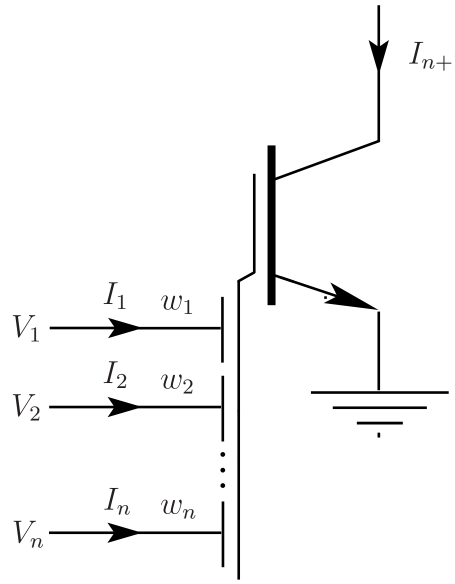


Figure 1.1. Symbol for a n-input multiple-input translinear element (MITE).

1.2 The Multiple-Input Translinear Element

The multiple-input translinear element (MITE) is a generalization of the BJT introduced by Bradley Minch [11,12]. A large class of linear and nonlinear systems can be implemented as MITE circuits [13,14,15,16,17,18,19,20,21,22]. In particular, all the functions that can be implemented by classical translinear circuits can be implemented using MITEs.

Definition 1.2.1 *The n–input MITE, as represented in Figure 1.1, is a $(n+1)$-port circuit element characterized by*

$$\begin{aligned} I_i &= 0 \qquad (i \in [1:n]) \\ I_{n+1} &= I_{\mathrm{s}} \exp\left[\kappa \frac{(w_1 V_1 + w_2 V_2 + \cdots w_n V_n)}{U_T}\right], \end{aligned} \tag{1.1}$$

where I_i and V_i are the port currents and port voltages. I_{s} is a pre-exponential scaling current and $U_T = \frac{kT}{q}$ is the thermal voltage. κ is a dimensionless scaling factor. The w_is are nonnegative weight coefficients, usually integers.

$w \triangleq \sum_{i=1}^{n} w_i$ is called the *fan-in* of the MITE [19]. A MITE with a fan–in of n is called a $n - MITE$. In particular, a $2 - MITE$ is a MITE with fan–in 2.

1.2.1 MITE implementations

As Equation (1.1) implies, a simple way to implement a MITE using the previously known translinear elements like a BJT or a subthreshold MOSFET is by using a multiple-input summer and the translinear element in cascade as shown in Figure 1.2(a) and (d). In the scheme in (a), the BJT (MOSFET) is in the common-emitter (common-source) configuration while in the scheme in (d), it is in a common-base (common-gate) configuration. The earliest circuits resembling MITE circuits used resistive dividers for the multiple-input summer in the configuration in Figure 1.2(a) [23, 24]. However, in current technologies, capacitor dividers are preferred due to area considerations. For the scheme in Figure 1.2(a), different configurations can be generated depending upon whether the summer is passive or active as shown in Figure 1.2(b) and (c), respectively. A popular implementation is the cascoded floating-gate MOSFET used in a common source configuration in Figure 1.3. The cascode transistor increases the output resistance of the floating-gate MOSFET and also helps in programming the floating-gate MOSFET [25]. An exemplary implementation of Figure 1.2(d) is shown in Figure 1.2(e) [26]. As can be seen from the implementations, the summer performance is considerably improved when the capacitances used in the summer are made of unit capacitances. Hence, the weight coefficients in a MITE are usually restricted to be nonnegative integers.

1.3 Product–of–power–law Networks

A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is called an algebraic function if there is a polynomial p in $n+1$ real variables such that $p(y, x_1, x_2, \ldots, x_n) = 0$, where $x_1, x_2, \ldots, x_n \in \mathbb{R}$ and $y = f(x_1, x_2, \ldots, x_n)$. If f is expressible by radicals, then it can be computed in a finite number of steps using only sums, products, and by raising a variable to known rational powers. If such is not the case, then y has to be found from $p(y, x_1, \ldots, x_n) = 0$ by means of feedback, which again involves the basic operations of sums, products, and powers. In translinear circuits, a sum of two or more currents is obtained using Kirchoff's current law (KCL) by simply joining the respective wires. The difference of two currents is obtained through a current mirror. Hence, only a product of variables raised to different powers is needed. The relevant MITE network implementing this is now discussed. The treatment given here

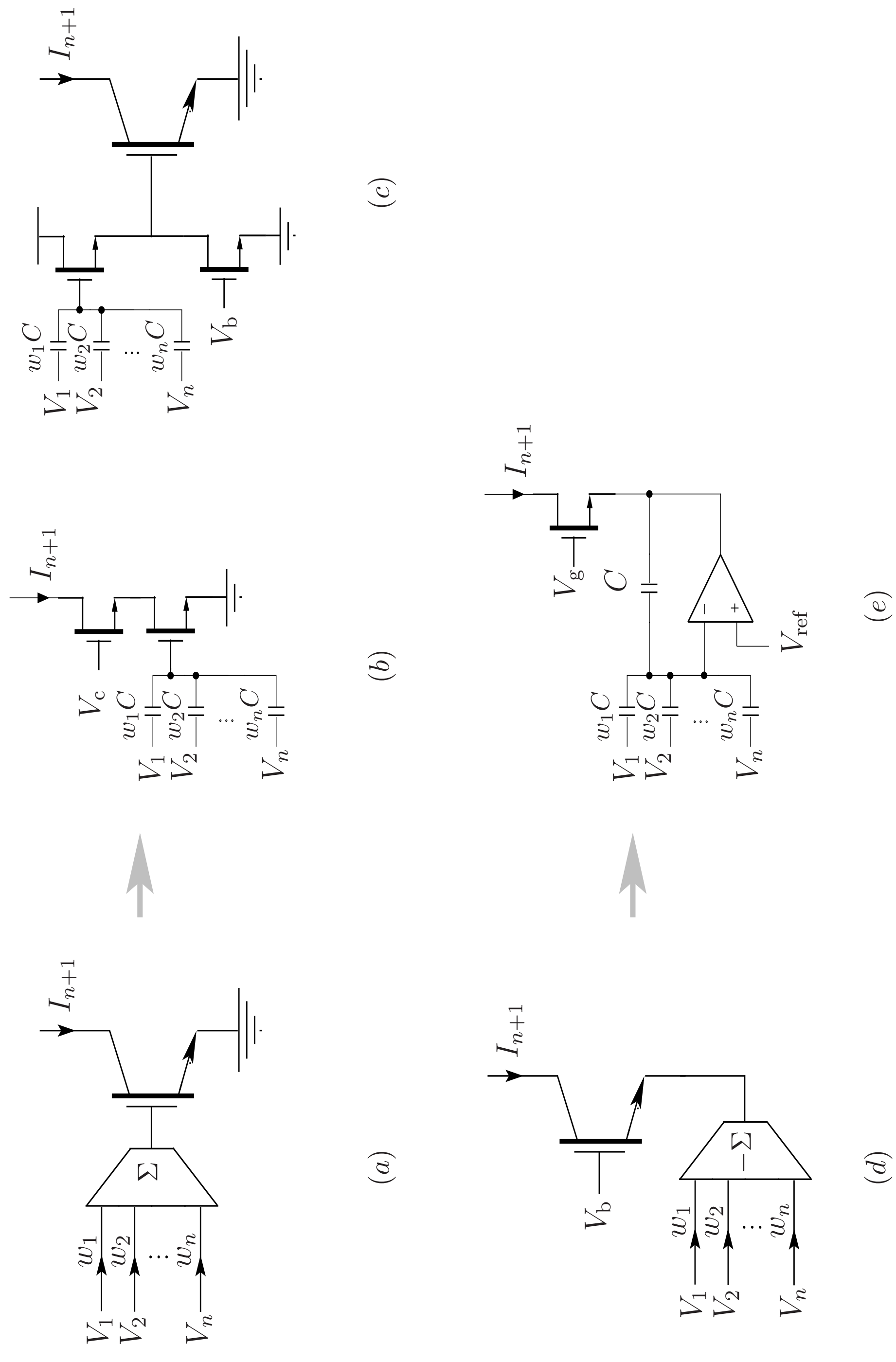


Figure 1.2. Schemes for constructing a n-input MITE. In the scheme in (a), the BJT (MOSFET) is in the common-emitter (common-source) configuration while in the scheme in (d), it is in a common-base (common-gate) configuration. Implementations of (a) are shown in (b) and (c), the former with a passive summer and the latter with an active summer. An implementation of (d) is shown in (e).

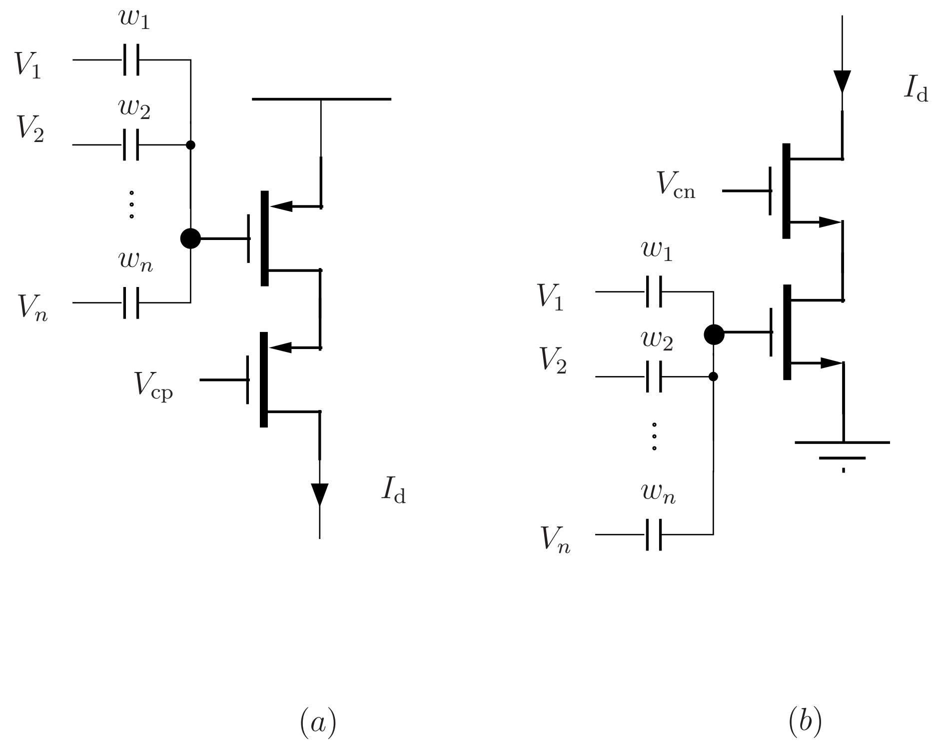


Figure 1.3. Cascoded floating-gate implementations of MITEs. The PFET implementation in (a) is the practical one; the NFET implementation in (b) is often used for illustrative purposes.

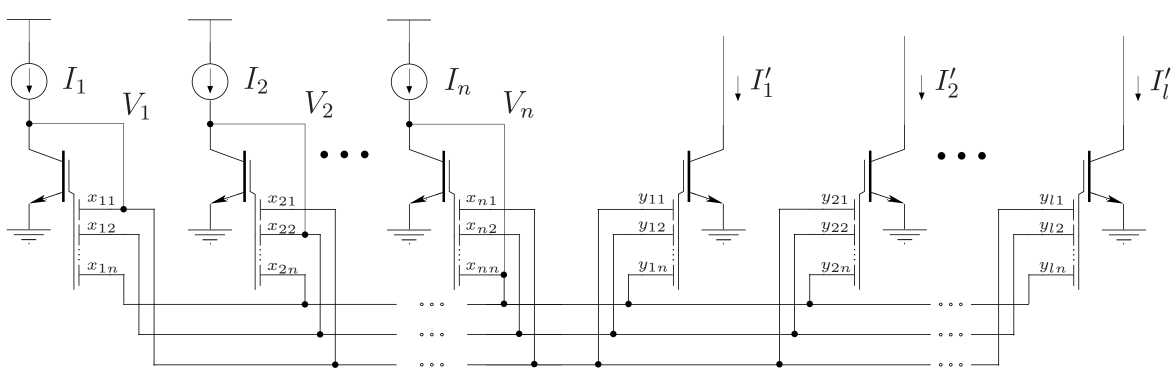


Figure 1.4. The general form of the MITE network implementing a POPL function. The output currents are a product of the input currents raised to different powers.

follows [12]. The mathematical notation used in the following as well as in the remainder of the thesis is described in the last section of the thesis.

Consider the MITE network in Figure 1.4. The weight coefficient matrices $X = [x_{ij}]$ and $Y = [y_{ij}]$ are called the input and output connectivity matrices, respectively. Using Equation (1.1), the relationship between $\mathbf{I} = [I_k], \mathbf{I}' = [I'_k]$, and $\mathbf{V} = [V_k]$ is arrived at. The parameters I_{s}, and κ are the same for all the MITEs in the circuit. It is clear that

$$\begin{aligned} \log\left\{\frac{I_q}{I_{\mathrm{s}}}\right\} &= \frac{\kappa}{U_{\mathrm{T}}}\sum_{k=1}^{n} x_{qk}V_k \quad q \in [1:n] \\ \log\left\{\frac{I'_p}{I_{\mathrm{s}}}\right\} &= \frac{\kappa}{U_{\mathrm{T}}}\sum_{k=1}^{n} y_{pk}V_k \quad p \in [1:l], \end{aligned} \tag{1.2}$$

which can be written in matrix form as

$$\begin{aligned} \log\left\{\frac{\mathbf{I}}{I_{\mathrm{s}}}\right\} &= \frac{\kappa}{U_{\mathrm{T}}}X\mathbf{V} \\ \log\left\{\frac{\mathbf{I}'}{I_{\mathrm{s}}}\right\} &= \frac{\kappa}{U_{\mathrm{T}}}Y\mathbf{V} \end{aligned} \tag{1.3}$$

It follows that if the input connectivity matrix X is invertible,

$$\log\left\{\frac{\mathbf{I}'}{I_{\mathrm{s}}}\right\} = YX^{-1}\log\left\{\frac{\mathbf{I}}{I_{\mathrm{s}}}\right\} \tag{1.4}$$

Define $\Lambda = YX^{-1}$. Removing the logarithms from Equation (1.4), it is clear that

$$\begin{aligned} \frac{I'_p}{I_{\mathrm{s}}} &= \prod_{q=1}^{n}\left\{\frac{I_q}{I_{\mathrm{s}}}\right\}^{\Lambda_{pq}} \\ I'_p &= I_{\mathrm{s}}^{1-\sum_{q=1}^{n}\Lambda_{pq}}\prod_{q=1}^{n} I_q^{\Lambda_{pq}} \end{aligned} \tag{1.5}$$

If X, Y are such that $\Lambda = YX^{-1}$ satisfies, for each p, $\sum_{q=1}^{n}\Lambda_{pq} = 1$ that can be written compactly as

$$\Lambda\mathbf{1}_n = \mathbf{1}_l, \tag{1.6}$$

then it can be concluded that

$$I'_p = \prod_{q=1}^{n} I_q^{\Lambda_{pq}} \tag{1.7}$$

Definition 1.3.1 *A MITE network as in Figure 1.4 characterized by a nonsingular input connectivity matrix and an output connectivity matrix is called a* product–of–power–law *(POPL) network. The output currents are products of the input currents raised to different powers as shown in Equation* (1.7)*.*

It should be noted that the functional relationship is independent of I_{s}, κ, and U_{T} and hence is independent of temperature as long as the assumption that I_{s} and κ are the same for all the MITEs is satisfied. Conditions under which this assumption holds will be discussed now.

The analysis of floating-gate MOSFETs in the subthreshold region relevant to MITE networks is done in [11,12]. For the NFET floating-gate shown in Figure 1.3(b), the current I_{d}, neglecting the dependence on the drain voltage, is given by

$$I_{\mathrm{d}} = I_0 \exp\left\{\frac{\kappa' Q}{C'_{\mathrm{T}} U_{\mathrm{T}}}\right\} \exp\left\{\frac{\kappa'}{C'_{\mathrm{T}} U_{\mathrm{T}}} \sum_{k=1}^{n} C_k V_k\right\}, \tag{1.8}$$

where C_k is the floating-gate capacitance connected between V_k and the floating gate. $\kappa' = C_{\mathrm{ox}}/(C_{\mathrm{ox}} + C_{\mathrm{dep}})$, C_{ox} and C_{dep} being the oxide capacitance and the depletion-layer capacitance of the MOSFET, respectively. Q is the charge on the floating node and C'_{T} is given by

$$C'_{\mathrm{T}} \triangleq \frac{C_{\mathrm{ox}} C_{\mathrm{dep}}}{C_{\mathrm{ox}} + C_{\mathrm{dep}}} + C_{\mathrm{b}} + \sum_{k=1}^{n} C_k + C_{\text{fg-s}} + C_{\text{fg-d}} + \kappa' C_{\text{fg-d}}, \tag{1.9}$$

where $C_{\mathrm{b}}, C_{\text{fg-s}}$, and $C_{\text{fg-d}}$ are the parasitic capacitances coupling onto the floating gate from the substrate, source, and the drain, respectively. We define $w_k \triangleq C_k/C$, where C is some reference capacitance and is commonly the unit capacitance that each of the floating-gate capacitances are made up of. Comparing Equation (1.8) with Equation (1.1), it is clear that $I_{\mathrm{s}} = I_0 \exp\left\{\frac{\kappa' Q}{C'_{\mathrm{T}} U_{\mathrm{T}}}\right\}$ and $\kappa = \kappa' Q/(C'_{\mathrm{T}} U_{\mathrm{T}})$.

In order that all the MITEs in Figure 1.4 have the same I_{s}, the charge Q on the floating gate needs to be controllable. This is achieved by means of *programming* [25] a floating-gate MOSFET. Even if I_0 is not the same for all MITEs, Q can be changed to account for the error so that I_{s} is made the same.

It is clear that κ depends on C'_{T} which in turn depends on the floating-gate capacitances C_k. Assuming that the parasitic capacitances are equal for each MITE, it follows from Equation (1.9) that for κ to be the same for all MITEs, the values of $\sum_{k=1}^{n} C_k$ should be the same. If the reference capacitance C is taken to be the same for all MITEs, the condition that $\sum_{k=1}^{n} w_k$ should be the same for all MITEs is obtained. By definition, this is the fan-in of each MITE. Hence, it can be concluded that for κ to be the same for all MITEs in Figure 1.4, the fan-in of all the MITEs should be the same.

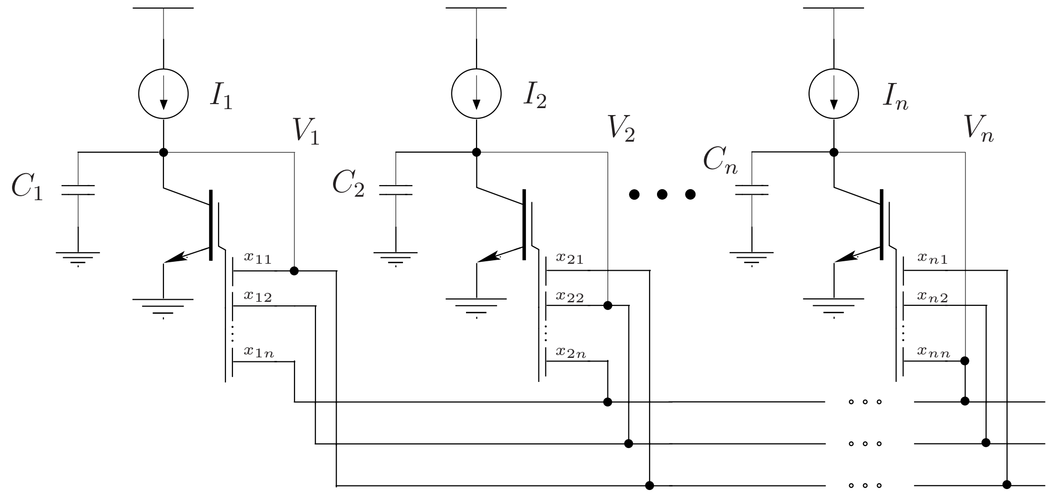


Figure 1.5. Stability analysis of the POPL network. Capacitances are attached from each node V_i to ground. The currents through the gates of all MITEs are neglected.

Definition 1.3.2 *[11]. The MITE network in Figure 1.4 is said to have a* balanced fan-in *or simply* is balanced *if the fan-in $\sum_k w_k$ is the same for all MITEs in the network. The common fan-in of all the MITEs is called the* fan-in of the network.

The following theorem is proved in [17, 12]:

Theorem 1.3.1 (Balanced Fan-In Theorem) *If a POPL MITE network as shown in Figure 1.4 is balanced, then the power matrix Λ of the network satisfies Equation* (1.6), *viz.* $\Lambda \mathbf{1}_n = \mathbf{1}_l$.

1.3.1 Stability of the Operating Point of POPL Networks

The stability of a POPL network depends upon the position assumed for the parasitic capacitances. In practice, the floating-gate capacitances themselves, along with the parasitic gate–source capacitance are the dominating capacitances. The resulting conditions depend upon the value of the capacitances and the transconductances of the MITEs. The analysis becomes much simpler if the significant parasitic capacitances are taken to be situated between ground and the drain of each MITE. This analysis is done in [12]. An abridged version of the derivation along with a discussion of the implied conditions on the connectivity matrices of the POPL network is given in this section. The nonnegligible parasitic capacitances C_i are assumed to be present from each V_i in Figure 1.5 to ground. For the stability analysis, the currents through the input gates of the MITEs along with their output conductances are neglected. Because of this, the "output" MITEs do not contribute to the stability of the network. Small signal analysis gives the following:

$$g_{\mathrm{m}i}(\sum_{j=1}^{n} x_{ij}v_j) + sC_iv_i = 0 \quad i \in [1:n] \tag{1.10}$$

For a ideal MITE,

$$g_{\mathrm{m}} = \frac{\partial I}{\partial V} = \frac{\partial}{\partial V}\left(I_{\mathrm{s}}\exp(\frac{\kappa V}{U_{\mathrm{T}}})\right) = \frac{\kappa I}{U_{\mathrm{T}}} > 0$$

and hence the stability depends both on the input currents and the parasitics in the MITE network. Dividing each equation in Equation (1.10) by $g_{\mathrm{m}i}$ and writing the set of equations in matrix form, one gets $(X+sT)\mathbf{v} = 0$, where $\mathbf{v} = [v_i]$ and $T = \mathrm{diag}(\tau_1, \tau_2, \ldots, \tau_n)$ is defined by $\tau_i = C_i/g_{\mathrm{m}i}$. It should be noted that the diagonal matrix $T > 0$. The characteristic polynomial, as defined in [27], of this network is thus given by $\det(X + sT) = 0$. Since $\det(T) \neq 0$, the characteristic polynomial can be written as $\det(sI - (-T^{-1}X)) = 0$. For the network to be stable, the eigenvalues of $-T^{-1}X$ are therefore required to lie in the open left–half s–plane. Since the eigenvalues of $-M$ are the negatives of the eigenvalues of M, it can be concluded that the eigenvalues of $T^{-1}X$ are required to be in the open right–half plane. This condition is useful only if the values of C_i and $g_{\mathrm{m}i}$ are known. Since only those networks whose stability is independent of both these sets of parameters are important, the condition should be valid for all diagonal matrices $T > 0$. Thus the following theorem is arrived at:

Theorem 1.3.2 *A necessary condition for a POPL MITE network to be stable irrespective of the value of input currents and parasitic capacitances is that its input connectivity matrix should be D-stable.*

By definition, M is a D-stable matrix if DM has eigenvalues in the open right–half s-plane for all diagonal matrices $D \geq 0$.

1.3.2 Sensitivity Considerations

In this section, we consider the sensitivity of the power matrix Λ to the floating-gate capacitances determining the connectivity matrices X and Y. This section is a presentation of the analysis given in [12].

Each non-negative integral weight w_k attached to a voltage V_k in a MITE is given by $w_k = C_k/C$, where C_k is the floating-gate capacitance attached to V_k. C is usually the unit capacitance out of which C_k is composed of and is usually the same for all MITEs in a particular MITE network. In practice, each unit capacitance C has the value $C + \Delta C_{kj}$, where the ΔC_{kj}'s are independent identically distributed random variables with zero mean and standard deviation σ_C.

Hence the new weight w'_k can be given as

$$w'_k = \frac{C_k}{C} = \frac{\sum_{j=1}^{w_k}(C + \Delta C_{kj})}{C} = w_k + \frac{\Delta C_k}{C} \tag{1.11}$$

Here the random variable $\Delta C_k \triangleq \sum_{j=1}^{w_k} \Delta C_{kj}$ has zero mean and variance given by $\sum_{j=1}^{w_k} \sigma_C^2 = w_k\sigma_C^2$. The power matrix $\Lambda = YX^{-1}$ now changes to $(Y + \Delta Y)(X + \Delta X)^{-1}$.

$$\begin{aligned}\Delta\Lambda &= (Y + \Delta Y)(X + \Delta X)^{-1} - YX^{-1} \\ &= (\Delta Y - \Lambda\Delta X)(X + \Delta X)^{-1} \\ &\approx (\Delta Y - \Lambda\Delta X)X^{-1},\end{aligned} \tag{1.12}$$

where the approximation in the last step is valid if we assume that the random matrix ΔX is bounded and that σ_C is sufficiently small.

For evaluating the variance of $\Delta\Lambda$, the following simple result is useful:

Lemma 1.3.1 *If the elements of $A = [a_{ik}] \in \mathcal{M}_{p,q}$ and $B = [b_{kj}] \in \mathcal{M}_{q,r}$ are random variables such that for all $i \in [1 : p]$, $j \in [1 : r]$, $s, k \in [1 : q]$*

E1 a_{ik} and b_{sj} are independent

E2 For $s \neq k$, either a_{ik}, a_{is} and/or b_{kj}, b_{sj} have zero mean and are independent,

then $\mathbb{E}(C \circ C) = \mathbb{E}(A \circ A)\mathbb{E}(B \circ B)$, where $C = AB$, $\mathbb{E}(.)$ denotes the expectation value of the random variable or matrix in the parentheses, and $\circ$ denotes the Hadamard product or the element-wise product of matrices. It should be noted that E1 is satisfied if one of A or B is a constant matrix.

Proof : We will prove the lemma for the case when a_{ik} and a_{is} are independent and leave the similar other case to the reader. It therefore follows from E2 that $\mathbb{E}(a_{ik}a_{is}) = \mathbb{E}(a_{ik}^2)\delta_{ks}$.

Let $C = AB = [c_{ij}]$.

$$\begin{aligned}
\mathbb{E}(c_{ij}^2) &= \mathbb{E}\big[(\sum_{k=1}^{q} a_{ik}b_{kj})(\sum_{s=1}^{q} a_{is}b_{sj})\big] \\
&= \mathbb{E}\big[\sum_{k,s=1}^{q} a_{ik}a_{is}b_{kj}b_{sj}\big] \\
&= \sum_{k,s=1}^{q} \mathbb{E}\big[a_{ik}a_{is}\big]\mathbb{E}\big[b_{kj}b_{sj}\big] \text{ (from E1)} \\
&= \sum_{k,s=1}^{q} \mathbb{E}\big[a_{ik}^2\big]\delta_{ks}\mathbb{E}\big[b_{kj}b_{sj}\big] \\
&= \sum_{k=1}^{q} \mathbb{E}\big[a_{ik}^2\big]\mathbb{E}\big[b_{kj}^2\big] \quad \square
\end{aligned}$$

In $\Delta\Lambda = (\Delta Y - \Lambda\Delta X)X^{-1}$, E1 is obviously satisfied. E2 is satisfied because $\Delta y_{ik} - \sum_{t=1}^{n}\Lambda_{it}\Delta x_{tk}$ has zero mean and since distinct elements of Y and X are independent. It should also be noted that E1 and E2 are also satisfied in the product $\Lambda\Delta X$. The variance of $\Delta\Lambda$ is then given by

$$\begin{aligned}
\mathbb{E}(\Delta\Lambda \circ \Delta\Lambda) &= \mathbb{E}\big[(\Delta Y - \Lambda\Delta X) \circ (\Delta Y - \Lambda\Delta X)\big]\mathbb{E}(X^{-1} \circ X^{-1}) \\
&= \mathbb{E}\big[\Delta Y \circ \Delta Y - 2\Delta Y \circ (\Lambda\Delta X) + (\Lambda\Delta X) \circ (\Lambda\Delta X)\big](X^{-1} \circ X^{-1}) \\
&= \big[\mathbb{E}(\Delta Y \circ \Delta Y) - 2\mathbb{E}(\Delta Y) \circ \mathbb{E}(\Lambda\Delta X) + \mathbb{E}[(\Lambda\Delta X) \circ (\Lambda\Delta X)]\big](X^{-1} \circ X^{-1}) \\
&= \big[(\sigma_C^2/C^2)Y + (\Lambda \circ \Lambda)\mathbb{E}(\Delta X \circ \Delta X)\big](X^{-1} \circ X^{-1}) \\
&= \frac{\sigma_C^2}{C^2}\big[Y + (\Lambda \circ \Lambda)X\big](X^{-1} \circ X^{-1})
\end{aligned}$$

1.4 Previous synthesis methods

1.4.1 Synthesis of Static MITE Networks

The synthesis problem of POPL networks is the problem of finding suitable input and output connectivity matrices for a given power matrix Λ. The previous synthesis procedures for POPL networks are discussed in [12,19,17,18,21,22]. All of these concentrate on synthesizing each output equation in 1.7 separately. Two network transformations that are important in this regard are *consolidation* and *completion*.

1.4.1.1 Consolidation

Once a MITE network is found for each equation in a set of equations using the above methods, consolidation is used to remove redundant MITEs. This is done by identifying MITEs that have the same drain current and have identical input voltages connecting to their gates.

1.4.1.2 Completion

Completion is a process of transforming a MITE network that is not balanced into a balanced one. This is done in [12] by finding the MITE with the maximum fan–in and then adding enough weights to other MITEs so that all of them have the same fan–in. The extra weights are typically connected to one of the controlling voltages that are already present in the MITE network.

A few points are to be noted in this regard:

1. The fan–in of the balanced MITE network arrived at after completion using the procedures in [12, 19] usually cannot be lesser than that of the MITE with the largest fan–in in the unbalanced network. This will be improved upon by the generalized completion theorem in Chapter 3.

2. According to the completion theorem in [19,22], the extra controlling voltage obtained during the process of adding weights to the MITEs can be connected to any of the controlling voltages already present in the unbalanced MITE network. It will be shown in Chapter 2 that this can sometimes lead to multiple operating points.

If consolidation is not possible for all voltages, then the final network has copies of the input currents flowing through different MITEs and then the procedure is not optimal with respect to the number of MITEs. On the other hand, these methods can potentially reduce the fan–in, and it follows from [19,12] that it can be reduced to the minimum possible value of 2. However, no procedure has been suggested to minimize the number of MITEs once the fan–in is fixed at some value.

A brief discussion of these synthesis methods now follows. The description is considerably simpler if these methods are presented using the formulation of the synthesis problem of POPL networks to be derived in Chapter 3. The reader is asked to refer to Section 3.3 in order to understand the discussion below.

As these methods are applicable without consolidation to the single–output case alone, we can assume that the power matrix Λ is a $1 \times n$ row-vector. Hence, the translinear loop matrix $A = [a_i] = [\Lambda \quad -1]$ is a $1 \times (n+1)$ row-vector. In Chapter 3, it is shown that the solution networks we are searching for are described by the *connectivity matrix* Z satisfying $AZ = 0$, which in this case is simply $Z = \left[\begin{smallmatrix} X \\ Y \end{smallmatrix}\right]$. Writing Z in terms of its columns as $Z = [\mathbf{z}_1 \ \mathbf{z}_2 \ldots \mathbf{z}_n]$, it follows that each column of Z needs to found amongst the solutions of the linear diophantine equation $A\mathbf{z} = 0$, where $\mathbf{z} \in \mathbb{N}^{n+1}$.

The synthesis strategies in [12] search for solutions for $A\mathbf{z} = 0$ that have exactly two nonzero entries. If $\mathbf{z} = [z_i]$ and if z_s and z_t are the only nonzero components of $\mathbf{z}$ ($s, t \in [1 : n+1]$), then we have $a_s z_s + a_t z_t = 0$. Since $\mathbf{z}$ is a nonnegative vector, it follows that a_s and a_t must necessarily have opposite signs. If we multiply A by a suitable integer to make all of its elements integers, then $z_s = \mathrm{lcm}(|a_s|, |a_t|)/|a_s|$ and $z_t = \mathrm{lcm}(|a_s|, |a_t|)/|a_t|$ are the two basic solutions of $a_s z_s + a_t z_t = 0$ with every other solution being an integer multiple of this. Hence, we define $\hat{\mathbf{z}}_{s,t} \in \mathbb{N}^{n+1}$ as

$$[\hat{\mathbf{z}}_{s,t}]_j = \begin{cases} \mathrm{lcm}(|a_s|, |a_t|)/|a_s| & \text{if } j = s \\ \mathrm{lcm}(|a_s|, |a_t|)/|a_t| & \text{if } j = t \\ 0 & \text{if } j \in [1 : n+1] \setminus \{s, t\} \end{cases} \tag{1.13}$$

Also, let $\mathcal{N} \triangleq \{i \in [1 : n+1] \ |\Lambda_i > 0\}$ and $\mathcal{D} \triangleq \{i \in [1 : n+1] \ |\Lambda_i < 0\}$

1.4.1.3 Two-layer networks synthesis procedure

1. First construct $Z = [\mathbf{z}_1 \ \mathbf{z}_2 \ldots \mathbf{z}_n]$ with

$$\mathbf{z}_i = \begin{cases} \hat{\mathbf{z}}_{i,n+1} & \text{if } i \in \mathcal{N} \\ \hat{\mathbf{z}}_{i,t_i} & \text{if } i \in \mathcal{D}, \end{cases} \tag{1.14}$$

   where t_i is chosen so that $t_i \in \mathcal{N}$ for every $i \in \mathcal{D}$.

2. Let $W = \|\sum_{i=1}^{n} \mathbf{z}_i\|_\infty$, which is simply the maximum row sum matrix norm of a matrix [28]. This is physically the fan–in of the MITE with the largest fan–in in the

network. The MITE network is now completed so that the *fan–in of the network* is W. Let k be the index of the controlling voltage to which the extra weights are connected in each MITE. The final connectivity matrix $\tilde{Z}$ can be written as $\tilde{Z} = [\tilde{\mathbf{z}}_1 \;\; \tilde{\mathbf{z}}_2 \ldots \tilde{\mathbf{z}}_n]$ with

$$\tilde{\mathbf{z}}_i = \begin{cases} \mathbf{z}_i & \text{if } i \neq k \\ W\mathbf{1}_n - \sum_{s=1}^{n} \mathbf{z}_s & \text{otherwise} \end{cases} \tag{1.15}$$

1.4.1.4 Cascade networks

First, the inputs are renumbered such that the elements of $\mathcal{N}$ are less than those in $\mathcal{D}$. Within $\mathcal{N}$ and $\mathcal{D}$ themselves, the indices can be arranged randomly. Therefore, let $\mathcal{N} = [1 : k]$ and $\mathcal{D} = [k+1 : n]$, where $k \leq n$.

1. Define $\mathbf{z}_1 = \hat{\mathbf{z}}_{1,n+1}$
2. $i := 1$. While $i \leq \min(k, n-k)$, do:
   - $\mathbf{z}_{i+k} = \hat{\mathbf{z}}_{i+k,i}$
   - If $i+1 \leq k$, $\mathbf{z}_{i+1} = \hat{\mathbf{z}}_{i+1,i+k}$
   - $i := i+1$
3. If $2k < n$, then for all i such that $k < i \leq n-k$, define $\mathbf{z}_{i+k} = \hat{\mathbf{z}}_{i+k,1}$.
4. if $2k > n$, then for all i such that $n-k+1 < i \leq k$, define $\mathbf{z}_i = \hat{\mathbf{z}}_{i,n+1}$.
5. Let $W = \|\sum_{i=1}^{n} \mathbf{z}_i\|_\infty$. The MITE network is now completed so that the *fan–in of the network* is W. Let s be the index of the controlling voltage to which the extra weights are connected in each MITE. The final connectivity matrix $\tilde{Z}$ can be written as $\tilde{Z} = [\tilde{\mathbf{z}}_1 \;\; \tilde{\mathbf{z}}_2 \ldots \tilde{\mathbf{z}}_n]$ with

$$\tilde{\mathbf{z}}_i = \begin{cases} \mathbf{z}_i & \text{if } i \neq s \\ W\mathbf{1}_n - \sum_{j=1}^{n} \mathbf{z}_j & \text{otherwise} \end{cases} \tag{1.16}$$

The cascade network can be used to show the existence of a 2-MITE implementation for any POPL equation with the assumption that any number of copies of input and output

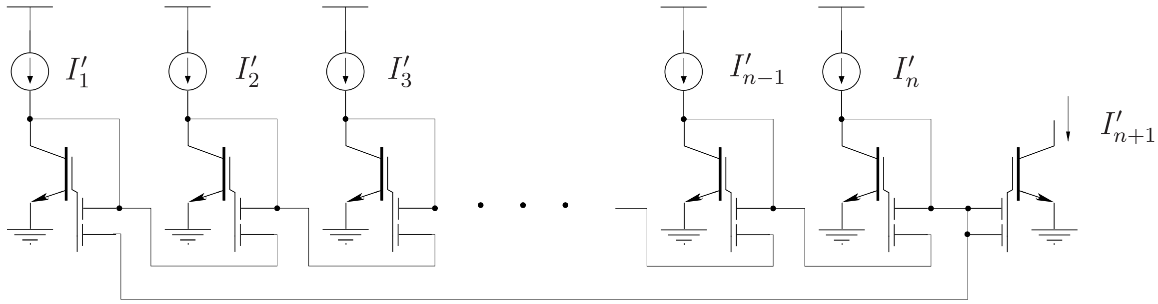


Figure 1.6. The cascade network implementing the equation $\prod_{k\ \mathrm{odd}}^{n'} I'_k = \prod_{k\ \mathrm{even}}^{n'} I'_k$.

currents can be used [29]. To show this, it is enough to prove that a single–output POPL equation can be implemented using 2-MITEs as the other equations can be implemented as separate 2-MITE networks using copies of input currents. By raising all the powers to a common integer, it is easy to see that we need to implement $\prod_{i=1}^{n+1} I_i^{a_i} = 1$, where $\sum a_i = 0$ and the a_i's are integers. By expanding $I_i^{a_i}$ as $\prod_{j=1}^{a_i} I_i$, if $a_i > 0$, and as $\prod_{j=1}^{|a_i|} I_i^{-1}$ if $a_i < 0$, the equation can be reduced to the form $\prod_{\substack{k=1 \\ k \text{ odd}}}^{n'} I'_k = \prod_{\substack{k=1 \\ k \text{ even}}}^{n'} I'_k$ which is the standard translinear loop equation form. The circuit shown in Figure 1.6 implements this equation for the case when $a_{n+1} = \pm 1$. It should be noted that for other values of a_{n+1}, current mirrors need to be used to ensure that the these currents match the output current.

1.4.2 Synthesis of Dynamic MITE Networks

Let the dynamical system to be implemented be given by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= g(\mathbf{x}(t), \mathbf{u}(t)) \end{aligned} \tag{1.17}$$

where $\mathbf{u}(t)$ is the input to the system, $\mathbf{x}(t)$ is the *state*, and $\mathbf{y}(t)$ is the output of the system. The previously existing synthesis procedures for dynamic systems using MITEs [30, 16, 22, 31, 32, 33] are discussed in brief below.

1.4.2.1 Exponential transformation

The variables $\mathbf{x}$, $\mathbf{u}$, and $\mathbf{y}$ are first scaled so that they can be written as the ratio of currents $\mathbf{I}_x = [I_{xi}]$, $\mathbf{I}_u = [I_{ui}]$, and $\mathbf{I}_y = [I_{yi}]$ to some unit current. We will still refer to the right-hand side of the transformed system equations by f and g, even though the notation is exact only for a linear system. To implement dynamical systems, the existing methods all make use of the exponential state-space transformation. Here the idea is to transform the given system $\dot{\mathbf{I}}_x = f(\mathbf{I}_x, \mathbf{I}_u)$ by making a state variable change from I_{xi} to V_i through $I_{xi} = \alpha_i \exp(\beta_i V_i)$, where α_i and β_i do not depend on time and $\alpha_i > 0$. It should be noted that the variable I_{xi} is always constrained to be positive because of the nature of the transformation. If it is otherwise, assuming that $I_{xi}(t)$ is bounded below by $-I_\mathrm{a}$, we can apply the same exponential-state transformation to $I_{xi} + I_\mathrm{a}$. If I_{xi} is not bounded below or if the lower bound $-I_\mathrm{a}$ is not known, then one can split I_{xi} into two positive signals I_{xi}^+ and I_{xi}^- satisfying some *differential* constraint such as the geometric constraint [31, 32, 34] so that the number of state variables and equations is doubled. In other words, the two equations corresponding to the i^{th} equation $\dot{I}_{xi} = f_i(\mathbf{I}_x, \mathbf{I}_u)$ are found by solving

$$\begin{aligned} \frac{\mathrm{d}}{\mathrm{d}t}(I_{xi}^+ - I_{xi}^-) &= f_i(\mathbf{I}_x, \mathbf{I}_u) \\ \frac{\mathrm{d}}{\mathrm{d}t}(I_{xi}^+ I_{xi}^-) &= I_\mathrm{b}^2 - I_{xi}^+ I_{xi}^- \end{aligned} \tag{1.18}$$

for $\dot{I}_{xi}^+$ and $\dot{I}_{xi}^-$, where I_b is a time-independent positive current. Bidirectional input signals $I_{ui}(t)$ can again be appropriately shifted if it is bounded with a known bound. Otherwise, an *algebraic* geometric constraint can be used to split the input signal $I_{ui}(t)$ into two input signals $I_{ui}^+(t)$ and $I_{ui}^-(t)$ [3]:

$$\begin{aligned} I_{ui}^+(t) - I_{ui}^-(t) &= I_{u_i}(t) \\ I_{ui}^+(t) I_{ui}^-(t) &= I_\mathrm{b}^2 \end{aligned} \tag{1.19}$$

It should be noted that Equation (1.19) is implementable using MITEs, as will be shown in Chapter 6. From the discussion above, it is clear that it can be assumed, without loss of generality, that $\mathbf{I}_x(t)$ *and* $\mathbf{I}_u(t)$ *are always positive*. Hence, by noting that $\dot{I}_{xi} = \beta_i I_{xi} \dot{V}_i$, we get the set of equations $\beta_i \dot{V}_i = f_i(\mathbf{I}_x, \mathbf{I}_u)/I_{xi}$.

1.4.2.2 MITE implementation

For some appropriately chosen capacitor value C_i, we get $C_i \dot{V}_i = (C_i/\beta_i) f_i(\mathbf{I}_x, \mathbf{I}_u)/I_{xi}$. The signal $C_i \dot{V}_i$ represents the current through a capacitor of value C_i. If f_i is a polynomial function of $\mathbf{I}_x$ and $\mathbf{I}_u$, then we can always find functions $f_i^+(\mathbf{I}_x, \mathbf{I}_u)$ and $f_i^-(\mathbf{I}_x, \mathbf{I}_u)$ such that they satisfy $f_i = f_i^+ - f_i^-$ and are always positive for any $\mathbf{I}_x$ and $\mathbf{I}_u$ – this can be done by simply grouping the positive and negative monomials. The same is true when

$f_i(\mathbf{I}_x, \mathbf{I}_u) = p(\mathbf{I}_x, \mathbf{I}_u)/q(\mathbf{I}_x, \mathbf{I}_u)$, where p and q are polynomials and q is positive for the values of $\mathbf{I}_x$ and $\mathbf{I}_u$ that we are interested in. Two cases arise, depending upon whether β_i is positive or negative:

Case 1: $\beta_i > 0$

The current equation to be implemented is $C_i\dot{V}_i + (C_i/\beta_i)f_i^- = (C_i/\beta_i)f_i^+$. The synthesis of this equation is shown in Figure 1.7(a). The *noninverting output structure* obeys the relation

$$I_{xi} = I_{\mathrm{s}} \exp\left(\kappa(w_i V_i + w_i' V_{\mathrm{ref}})/U_{\mathrm{T}}\right), \tag{1.20}$$

where the weight w_i' attached to the time-independent voltage V_{ref} is chosen so that the MITE network is balanced. Clearly, $\alpha_i = I_{\mathrm{s}} \exp(\kappa w_i' V_{\mathrm{ref}}/U_{\mathrm{T}})$ and $\beta_i = \kappa w_i/U_{\mathrm{T}}$.

Case 2: $\beta_i < 0$

The current equation to be implemented is $C_i\dot{V}_i + (C_i/|\beta_i|)f_i^+ = (C_i/|\beta_i|)f_i^-$. The synthesis of this equation is shown in Figure 1.7(b). The *inverting output structure* is required to generate the negative β_i. To find β_i and α_i for this structure, we have the following equations:

$$\begin{aligned} I_{xi} &= I_{\mathrm{s}} \exp\left(\frac{\kappa}{U_{\mathrm{T}}}(w_i V_{\mathrm{int}} + w_i' V_{\mathrm{ref}})\right) \\ I_{\mathrm{b}} &= I_{\mathrm{s}} \exp\left(\frac{\kappa}{U_{\mathrm{T}}}(w_{i1} V_{\mathrm{int}} + w_{i1}' V_i)\right) \end{aligned} \tag{1.21}$$

Eliminating V_{int}, we get

$$I_{xi} = I_{\mathrm{b}}^{w_i/w_{i1}} I_{\mathrm{s}}^{1-w_i/w_{i1}} \exp\left(\frac{\kappa}{U_{\mathrm{T}}}\left(-\frac{w_i w_{i1}'}{w_{i1}} V_i + w_i' V_{\mathrm{ref}}\right)\right), \tag{1.22}$$

where the weights are as shown in Figure 1.7(b). The assumption that the MITE network is balanced leads to $w_i + w_i' = w_{i1} + w_{i1}'$. Clearly, $\alpha_i = I_{\mathrm{b}}^{w_i/w_{i1}} I_{\mathrm{s}}^{1-w_i/w_{i1}} \exp\left(\kappa w_i' V_{\mathrm{ref}}/U_{\mathrm{T}}\right)$ and $\beta_i = -\kappa w_i w_{i1}'/(w_{i1}U_{\mathrm{T}})$. In particular, if $w_i = w_{i1}$, it follows that $w_i' = w_{i1}'$, $\alpha_i = I_{\mathrm{b}} \exp\left(\kappa w_i' V_{\mathrm{ref}}/U_{\mathrm{T}}\right)$, $\beta_i = -\kappa w_i'/U_{\mathrm{T}}$, and

$$I_{xi} = I_{\mathrm{b}} \exp\left(\frac{\kappa w_i'}{U_{\mathrm{T}}}(-V_i + V_{\mathrm{ref}})\right). \tag{1.23}$$

It should be noted that the static translinear block in Figure 1.7 can use both $\mathbf{V} = [V_j]$ and $\mathbf{I}_x = [I_{xj}]$ to generate the desired output currents. Further, the output currents can be generated from a single connected static MITE block and not necessarily through two

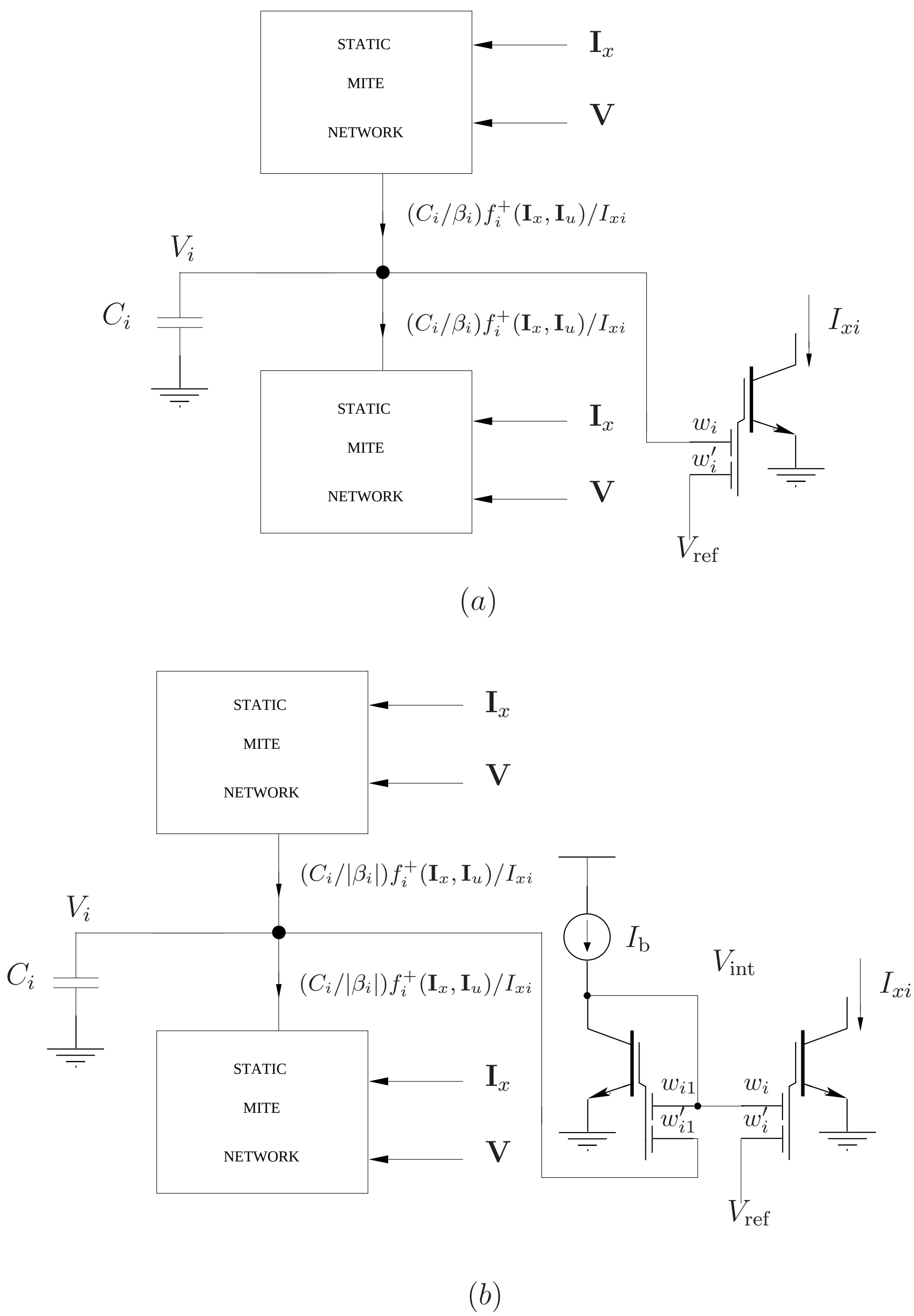


Figure 1.7. The MITE implementation of the i^{th} equation in the set of equations $\dot{\mathbf{I}_x} = f(\mathbf{I}_x, \mathbf{I}_u)$. The state variable I_{xi} is transformed into V_i through $I_{xi} = \alpha_i \exp(\beta_i V_i)$. The noninverting output structure for $\beta_i > 0$ is shown in (a) and the inverting output structure for $\beta_i < 0$ is shown in (b). The static MITE networks take as inputs the vector variables $\mathbf{I}_x$ and $\mathbf{V}$ and produces outputs $(C_i/\beta_i)f_i^+/I_{xi}$ and $(C_i/\beta_i)f_i^-/I_{xi}$.

disconnected blocks as shown in Figure 1.7. As an example, the synthesis of a MITE first-order lowpass filter, given in [30], is described below. The equation to be implemented is

$$\tau \dot{I}_y + \beta I_y = \alpha I_x. \tag{1.24}$$

If an inverting output structure is assumed with $w_i = w_{i1} = 1$ and a fan–in of 2, it follows that $I_y = I_\mathrm{b} \exp\left(\kappa(-V_y + V_\mathrm{ref})/U_\mathrm{T}\right)$. Hence, we get $\dot{I}_y = -I_y \dot{V}_y \kappa / U_\mathrm{T}$. Thus , we need to implement $C\dot{V}_y = -\left(CU_\mathrm{T}/(\tau\kappa)\right)\left(\alpha I_x / I_y\right) + \beta C U_\mathrm{T}/(\tau\kappa)$. Defining $I_{\tau 1} = \beta C U_\mathrm{T}/(\tau\kappa)$ and $I_{\tau 2} = \alpha C U_\mathrm{T}/(\tau\kappa)$, it is clear that the current equation that needs to be implemented is

$$C\frac{\mathrm{d}V_y}{\mathrm{d}t} + \frac{I_{\tau 2} I_x}{I_y} = I_{\tau 1}$$

A static MITE circuit is needed to implement

$$I_p = \frac{I_{\tau 2} I_x}{I_y} = \frac{I_{\tau 2}}{I_\mathrm{b}} \frac{I_x}{\exp(\kappa(V_\mathrm{ref} - V_y)/U_\mathrm{T}}.$$

A simple set of deductions from this equation leads us to the standard solution. We can choose I_b as any positive current we want, as long as it is time-independent. Hence, if $I_{\tau 2}$ is a *time-independent* current, it follows that we can choose $I_{\tau 2} = I_\mathrm{b}$. Secondly, if we feed the input current I_x to a diode-connected 2-MITE with V_x as the drain voltage and V_ref as the other input-gate voltage, then it implies that I_p is simply $I_\mathrm{s} \exp\left(\kappa(V_x + V_y)/U_\mathrm{T}\right)$, which is the drain current of a MITE with gate voltages V_x and V_y. The final lowpass filter structure is shown in Figure 1.8. Clearly, it obeys the equation

$$\frac{CU_\mathrm{T}}{\kappa}\dot{I}_y + I_{\tau 1} I_y = I_{\tau 2} I_x, \tag{1.25}$$

where $I_{\tau 2}$ is necessarily time-independent. Further, it should be noted that at no point did we need to restrict $I_{\tau 1}$, or equivalently β, to be independent of time. The fact that the current $I_{\tau 1}$ need not be constant for the equation to hold is important and will be used for the development of a new synthesis procedure for dynamic systems in Chapter 6.

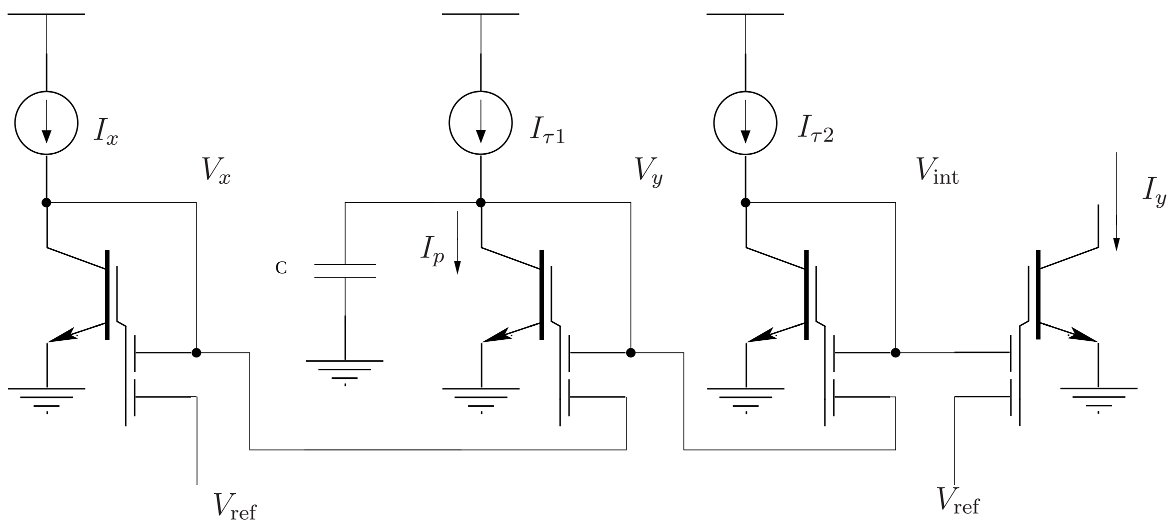


Figure 1.8. The standard MITE first–order lowpass filter. The filter obeys the equation $(CU_{\mathbf{T}})/\kappa \dot{I}_y + I_{\tau 1} I_y = I_{\tau 2} I_x$.

CHAPTER 2

CONDITIONS ON MATRICES ASSOCIATED WITH MITE CIRCUITS

The objective of this chapter is the derivation of certain conditions that will be imposed on different matrices associated with MITE networks for synthesis purposes. These conditions follow from a consideration of the deviation in the transfer characteristic of the practical MITE from the desired exponential behavior. These are divided into two cases, namely those conditions that originate from the static non-ideal behavior of the practical MITE and those that originate from dynamic non-ideal behavior of a practical MITE due to the presence of parasitic capacitances. The ideal MITE expression is valid only in the subthreshold saturation region. These networks typically have multiple feedback loops and hence, if not synthesized properly, will have multiple operating points not predicted by the ideal relationship. In particular, the output resistance of a MITE cannot be neglected *even if the output resistance is small or zero in the exponential region* if it is significant in other regions. Conditions on the topology of a general MITE network are presented that ensure that the operating point, if it exists, is unique. Hence, we find that the operating point predicted by the ideal MITE expression is the only one under these conditions. Besides the static characteristics, parasitic capacitances or the floating-gate capacitances themselves affect the stability of a static MITE circuit. In particular, for a POPL MITE network, conditions on the input connectivity matrix have been derived so that the equilibrium point is stable for all input currents.

2.1 Static Modeling of the Nonideal MITE

By definition, the ideal MITE, shown in Figure 2.1(a), is a $n+1$-port element with the constitutive equations:

$$\begin{aligned} I_i &= 0 \qquad (i \in [1:n]) \\ I_{n+1} &= I_{\mathrm{s}} \exp\left(\frac{\kappa}{U_{\mathrm{T}}}(w_1 V_1 + w_2 V_2 + \cdots + w_n V_n)\right), \end{aligned} \tag{2.1}$$

where κ is a dimensionless constant; I_{s} and U_{T} are scaling constants. A practical MITE implementation using a floating-gate MOSFET approximates a ideal MITE reasonably only when the MOSFET is in the subthreshold saturation region. Using only the ideal expression in Equation (2.1) for analysis has two disadvantages:

1. Circuits designed to have a unique operating point using the ideal model need not behave so in practice.

2. Circuits designed to have a monotonic input-output relationship using the ideal model need not behave so in practice.

The two criteria are related since the statement that there is an input-output relationship in a MITE circuit itself implicitly assumes that there is a unique output, and hence a unique operating point, for a given input. Also, sufficient conditions that imply the existence of a unique operating point in a circuit generally assume monotonicity of at least some of the blocks [35].

Hence, there is a need for a model of a MITE that covers all the regions where the floating-gate MOSFET might operate. Also, since a MITE has implementations other than the simple floating-gate one, our general model should model most of the these also. A general model of the MITE in Figure 2.1(a) taking into account the behavior in different regions of the MITE is the following, $w \triangleq \sum_{i=1}^{n} w_i$ being the fan–in of the MITE:

$$\begin{aligned} I_i &= 0 \qquad (i \in [1:n]) \\ I_{n+1} &= f(w_1 V_1 + w_2 V_2 + \cdots + w_n V_n, V_{n+1}), \end{aligned} \tag{2.2}$$

where the function $f : (0, wV_{\mathrm{DD}}) \times (0, V_{\mathrm{DD}}) \mapsto (0, \infty)$ is continuously differentiable and satisfies

$$g_{\mathrm{m}} \triangleq \frac{\partial f}{\partial x}(x, y) > 0 \quad g_{\mathrm{o}} \triangleq \frac{\partial f}{\partial y}(x, y) \geq 0 \tag{2.3}$$

for all $x \in (0, wV_{\mathrm{DD}})$ and $y \in (0, V_{\mathrm{DD}})$. The function g_{m} will be called as the *transconductance* of the MITE and g_{o} is clearly the *output conductance* of the MITE. These assumptions naturally follow from MOSFET modeling, if the MITE is considered as a voltage divider whose output is connected to the gate of a NFET. All the MITE circuits of interest here have PFETs with their source and bulk connected to V_{DD}, as shown in Figure 2.1(b). Such

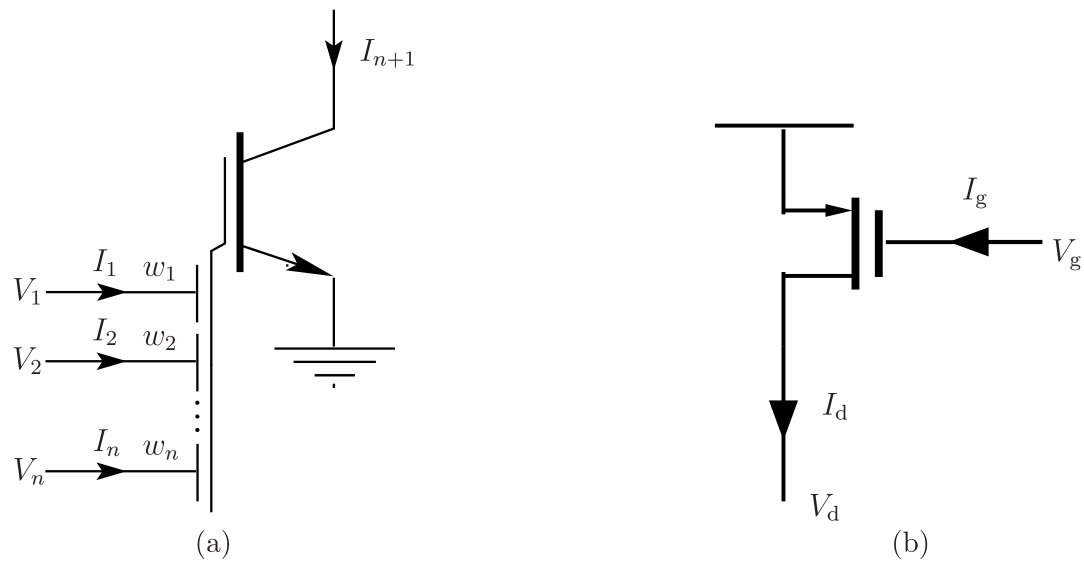


Figure 2.1. (a) Symbol for a n-input MITE. (b) Symbol of a PFET modeled by Equation (2.11)**. The same symbol is used for a cascoded PFET.**

PFETs will be modeled by

$$\begin{aligned} I_{\mathrm{g}} &= 0 \\ I_{\mathrm{d}} &= g(V_{\mathrm{g}}, V_{\mathrm{d}}), \end{aligned} \tag{2.4}$$

where the function $g : (0, V_{\mathrm{DD}}) \times (0, V_{\mathrm{DD}}) \mapsto (0, \infty)$ is continuously differentiable and satisfies

$$g_{\mathrm{m}} \triangleq -\frac{\partial g}{\partial x}(x, y) > 0 \quad g_{\mathrm{o}} \triangleq -\frac{\partial g}{\partial y}(x, y) \geq 0 \tag{2.5}$$

for all $x, y \in (0, V_{\mathrm{DD}})$. The functions g_{m} and g_{o} are clearly the transconductance and the output conductance of the PFET. Cascoded MITEs or PFETs have a similar characteristic and will be represented by the same symbols.

2.2 Mathematical Preliminaries

The following theorem is got from well-known ideas in [36, 37].

Theorem 2.2.1 *Let U be a convex subset of $\mathbb{R}^n$. Let $f : U \mapsto \mathbb{R}^n$ be a C^1 function satisfying the following condition:*

$$\begin{aligned} &\forall\, \mathbf{y}, \mathbf{z} \in U,\ \det(K(\mathbf{y}, \mathbf{z})) \neq 0 \\ &\textit{where } [K(\mathbf{y}, \mathbf{z})]_{ij} \triangleq \int_0^1 \frac{\partial f_i}{\partial x_j}((1-\alpha)\mathbf{y} + \alpha\mathbf{z})\mathrm{d}\alpha \\ &\textit{or, equivalently } K(\mathbf{y}, \mathbf{z}) = \int_0^1 J_f((1-\alpha)\mathbf{y} + \alpha\mathbf{z})\mathrm{d}\alpha \end{aligned} \tag{2.6}$$

Then f is injective i.e., one-one on U.

Proof If $\mathbf{z}, \mathbf{y} \in U$, then the line segment joining the two points is in U. Then, by the Fundamental Theorem of Calculus,

$$
\begin{aligned}
f_i(\mathbf{z}) - f_i(\mathbf{y}) &= \int_0^1 \frac{\mathrm{d}}{\mathrm{d}\alpha}(f_i((1-\alpha)\mathbf{y} + \alpha\mathbf{z}))\mathrm{d}\alpha \\
&= \int_0^1 \sum_{j=1}^{n} (z_j - y_j)\frac{\partial f_i}{\partial x_j}((1-\alpha)\mathbf{y} + \alpha\mathbf{z})\mathrm{d}\alpha \\
&= \sum_{j=1}^{n} [K(\mathbf{y}, \mathbf{z})]_{ij}(z_j - y_j)
\end{aligned}
$$

Hence, $f(\mathbf{z}) - f(\mathbf{y}) = K(\mathbf{y}, \mathbf{z})(\mathbf{z} - \mathbf{y})$. Since $K(\mathbf{y}, \mathbf{z})$ is an invertible matrix by the hypothesis of the theorem, $f(\mathbf{z}) = f(\mathbf{y})$ implies $\mathbf{y} = \mathbf{z}$. Therefore, f is injective on U. □

The theorem given below is proved by induction in [38].

Theorem 2.2.2 *For each positive integer n, the multiaffine function*

$$
\begin{aligned}
&c_0 + c_1 d_1 + c_2 d_2 + \cdots c_n d_n + c_{12} d_1 d_2 + \cdots c_{n-1,n} d_{n-1} d_n + \cdots + \\
&\qquad c_{i_1, i_2, \ldots, i_k} d_{i_1} d_{i_2} \cdots d_{i_k} + \cdots c_{1,2,\ldots,n} d_1 d_2 d_3 \cdots d_n \qquad (2.7)
\end{aligned}
$$

is nonzero for all positive values of the variables $d_1, d_2, \ldots, d_n$ if and only if at least one of the coefficients $c_{i_1, i_2, \ldots, i_k}$ is nonzero and all the nonzero coefficients have the same sign.

2.2.0.3 Some Definitions [39]

A matrix $M \in \mathcal{M}_n(\mathbb{R})$ is called a

1. *P-matrix* if all of its principal minors are positive.

2. P_0-*matrix* if all of its principal minors are nonnegative.

3. P_0^+-*matrix* if all of its k-by-k principal minors are nonnegative with at least one positive for each k.

4. *D-stable* matrix if DM has eigenvalues in the open right–half s-plane for all diagonal matrices $D \geq 0$.

The set of $n \times n$ real P-matrices, P_0-matrices, P_0^+-matrices, and D-stable matrices will be denoted by $\mathcal{P}_n(\mathbb{R})$, $\mathcal{P}0_n(\mathbb{R})$, $\mathcal{P}0_n^+(\mathbb{R})$, and $\mathcal{D}_n(\mathbb{R})$, respectively. If $\mathbb{F} \subseteq \mathbb{R}$, then $\mathcal{P}_n(\mathbb{F})$,

$\mathcal{P}0_n(\mathbb{F})$, $\mathcal{P}0_n^+(\mathbb{F})$, and $\mathcal{D}_n(\mathbb{F})$ are defined to be $\mathcal{P}_n(\mathbb{R}) \cap \mathcal{M}_n(\mathbb{F})$, $\mathcal{P}0_n(\mathbb{R}) \cap \mathcal{M}_n(\mathbb{F})$, $\mathcal{P}0_n^+(\mathbb{R}) \cap \mathcal{M}_n(\mathbb{F})$, and $\mathcal{D}_n(\mathbb{R}) \cap \mathcal{M}_n(\mathbb{F})$, respectively. When $\mathbb{F}$ is not specified, it will be taken to refer to $\mathbb{R}$.

2.3 POPL Networks

The general POPL network [12] is shown in Figure 1.4. It is clear that, since the output MITEs do not load the input MITEs, the uniqueness of the operating point is determined solely by the input side; i.e., by the input connectivity matrix $X = [x_{ij}]$ [12].

Before going into the analysis of this network using the general MITE model described in Section 2.1, let us analyze the circuit assuming that the input currents are restricted so that the MITEs in the input section of the POPL network are in the subthreshold region. Let us further assume that these MITEs are floating-gate NFETs that are not cascoded. It should be noted that since the drain voltage of each MITE is determined by the *circuit itself* by feedback, there is always the possibility of the MITE being in the nonsaturation region, unlike in the case of a normal MOSFET. From [12, 11], the drain current of the MITE can be written as

$$I_{\mathrm{d}} = \frac{W}{L} I_0 \exp\left\{\frac{\kappa' Q}{C_{\mathrm{T}} U_{\mathrm{T}}}\right\} \exp\left\{\sum_{k=1}^{n} \frac{\kappa' C_k}{C_{\mathrm{T}}} \frac{V_k}{U_{\mathrm{T}}}\right\} \left[\exp\left\{\frac{V_{\mathrm{d}}}{V_{\mathrm{A}}}\right\} \left(1 - \exp\left\{-\frac{V_{\mathrm{d}}}{U_{\mathrm{T}}}\right\}\right)\right] \tag{2.8}$$

where the constants are defined in the same way as in Equation (1.8) in Section 1.3. C_{T} is given by

$$C_{\mathrm{T}} \triangleq \frac{C_{\mathrm{ox}} C_{\mathrm{dep}}}{C_{\mathrm{ox}} + C_{\mathrm{dep}}} + C_{\mathrm{b}} + \sum_{k=1}^{n} C_k + C_{\text{fg-s}} + C_{\text{fg-d}}, \tag{2.9}$$

Comparing Equation (2.8) with Equation (2.1), we find that the nonideal floating-gate NFET MITE can be modeled as

$$\begin{aligned} I_i &= 0 \qquad (i \in [1:n]) \\ I_{n+1} &= I_{\mathrm{s}} \exp\left(\frac{\kappa}{U_{\mathrm{T}}}(w_1 V_1 + w_2 V_2 + \cdots + w_n V_n)\right) \left[\exp\left\{\frac{V_{n+1}}{V_{\mathrm{A}}}\right\} \left(1 - \exp\left\{-\frac{V_{n+1}}{U_{\mathrm{T}}}\right\}\right)\right] \end{aligned} \tag{2.10}$$

The current sources shown are usually PFETs that are cascoded or otherwise with the gate voltage fixed at some value in the range $(0, V_{\mathrm{DD}})$ and the source connected to V_{DD}. For the sake of simplicity, we will assume that the PFETs are not cascoded and that their constitutive equation in the subthreshold region is given by

$$\begin{aligned} I_{\mathrm{g}} &= 0 \\ I_{\mathrm{d}} &= I_{\mathrm{sp}} \exp(\frac{\kappa_p(V_{\mathrm{DD}} - V_{\mathrm{g}})}{U_{\mathrm{T}}})\big[\exp(\frac{V_{\mathrm{DD}} - V_{\mathrm{d}}}{V_{\mathrm{Ap}}})\{1 - \exp(-\frac{V_{\mathrm{DD}} - V_{\mathrm{d}}}{U_{\mathrm{T}}})\}\big] \end{aligned} \tag{2.11}$$

If the gate voltage of the current source I_i is $V_{\mathrm{g}i}$, then KCL gives

$$\begin{aligned} & I_{\mathrm{s}} \exp(\frac{\kappa}{U_{\mathrm{T}}}(\sum_{j=1}^{n} x_{ij} V_i))\Big[\exp\left\{\frac{V_i}{V_{\mathrm{A}}}\right\}\big(1 - \exp\left\{-\frac{V_i}{U_{\mathrm{T}}}\right\}\big)\Big] \\ = \quad & I_{\mathrm{sp}} \exp(\frac{\kappa_p(V_{\mathrm{DD}} - V_{\mathrm{g}i})}{U_{\mathrm{T}}})\big[\exp(\frac{V_{\mathrm{DD}} - V_i}{V_{\mathrm{Ap}}})\{1 - \exp(-\frac{V_{\mathrm{DD}} - V_i}{U_{\mathrm{T}}})\}\big] \end{aligned} \tag{2.12}$$

Taking logarithms on both sides and noting that $I_{\mathrm{sp}} \exp(\frac{\kappa_p(V_{\mathrm{DD}} - V_{\mathrm{g}i})}{U_{\mathrm{T}}})$ is simply the ideal value I_i of the i^{th} current source, we get

$$\sum_{j=1}^{n} x_{ij} V_j + h_i(V_i) = b_i \quad (i \in [1:n]) \tag{2.13}$$

where $b_i = U_{\mathrm{T}}/\kappa \log(I_i/I_{\mathrm{s}})$ and h_i is defined by

$$h_i(V_i) = \frac{U_{\mathrm{T}}}{\kappa} \log\left[\frac{\exp\{V_i/V_{\mathrm{A}}\}\big(1 - \exp\{-V_i/U_{\mathrm{T}}\}\big)}{\exp\{(V_{\mathrm{DD}} - V_i)/V_{\mathrm{Ap}}\}\big(1 - \exp\{-(V_{\mathrm{DD}} - V_i)/U_{\mathrm{T}}\}\big)}\right] \tag{2.14}$$

By noting that each factor in the numerator of the function inside the logarithm is an increasing function of V_i and that the factors of the denominator are decreasing functions of V_i, we can conclude that h_i is an strictly increasing function of V_i. Also, it should be noted that h_i is a function of only V_i. Equation (2.13) can be written in matrix form as

$$XV + H(V) = B \tag{2.15}$$

where the variable $V = [V_i]$ is taken from the set $(0, V_{\mathrm{DD}})^n$ and $H(V) = [h_i(V_i)]$. This equation is popular in nonlinear circuit theory as evidenced by the number of papers dealing with it in [35]. However, the case dealt with usually is the one where the solution is searched on $\mathbb{R}^n$, where Palais' theorem [38] is used to prove the existence and uniqueness of the solution. Clearly, this is not directly applicable here. It is shown in Corollary 1 of Theorem 3 in [40] that Equation (2.15) has an unique solution if X is a P_0 matrix, if any solution exists at all (Though we do not deal with it here, the existence of an operating point in this case can be proved using the results of [41]). A simple proof, from [40], follows:

Theorem 2.3.1 *Let $X \in \mathcal{M}_n(\mathbb{R})$ be a P_0 matrix. Let $h_i : (0, V_{\mathrm{DD}}) \mapsto \mathbb{R}$ be strictly monotonically increasing for all $i \in [1:n]$. Let $H : (0, V_{\mathrm{DD}})^n \mapsto \mathbb{R}^n$ be defined by $[H(V)]_i = h_i(V_i)$ and let B be any vector in $\mathbb{R}^n$. Then, the equation $XV + H(V) = B$ has at most one solution in $(0, V_{\mathrm{DD}})^n$.*

Proof If, on the contrary, two solutions V and $\hat{V}$ exist, then it follows that $X(V - \hat{V}) + H(V) - H(\hat{V}) = 0$. The i^{th} element of $H(V) - H(\hat{V})$ is $h_i(V_i) - h_i(\hat{V}_i)$. Now, if $V_i \neq \hat{V}_i$, then $d_i = [h_i(V_i) - h_i(\hat{V}_i)]/(V_i - \hat{V}_i)$ is a positive real number and if $V_i = \hat{V}_i$, $h_i(V_i) - h_i(\hat{V}_i) = d_i(V_i - \hat{V}_i)$ holds for any positive d_i. Hence, it follows that $H(V) - H(\hat{V}) = D(V - \hat{V})$ for some diagonal matrix $D > 0$, which implies that $(X + D)(V - \hat{V})$. However, $X + D$ is nonsingular since X is a P_0 matrix [40], which contradicts the assumption of two distinct solutions. □

It should be noted that the form $XV + H(V) = B$ is arrived at only because of the assumption that the drain current current expression of the i^{th} MITE is of the form $\exp\{\alpha \sum_j x_{ij} V_j\} f_i(V_i)$. There is no reason that this should be the case, and hence the need for the general models given in Equation (2.2) and Equation (2.4). We now prove that even using these general models, for uniqueness of the operating point, it suffices that X be a P_0 matrix and a nonsingular matrix.

In the general case, KCL gives us the following:

$$f_i\Big(\sum_{j=1}^{n} x_{ij} V_j, V_i\Big) - g_i(V_{\mathrm{g}i}, V_i) = 0 \quad (i \in [1:n]), \tag{2.16}$$

where f_i represents the drain current for the i^{th} MITE and g_i is the drain current of the i^{th} current source as discussed in Section 2.1. Since the circuit is to operate with the MITEs in the region of near-exponential behavior and the current sources with large output resistance, it is assumed that the $V_{\mathrm{g}i}$ are such that there is a set $\{V_i\}$ satisfying the above equation in the desired region. Now, to show that this solution is unique, it is enough to show that the function $F : (0, V_{\mathrm{DD}})^n \mapsto \mathbb{R}^n$ is injective, where

$$F_i(V_1, V_2, \ldots, V_n) = f_i\Big(\sum_{j=1}^{n} x_{ij} V_j, V_i\Big) - g_i(V_{\mathrm{g}i}, V_i)$$

The Jacobian matrix of F is given by

$$\frac{\partial F_i}{\partial V_j} = g_{\mathrm{m}i} x_{ij} + g_{\mathrm{on}i}\delta_{ij} + g_{\mathrm{op}i}\delta_{ij},$$

where $g_{\mathrm{m}i}$ and $g_{\mathrm{on}i}$ are the transconductance and the output conductance of the i^{th} MITE, respectively, as defined in Equation (2.3). The function $g_{\mathrm{op}i}$ is the output conductance of the PFET current source. It should be noted that the g_{m}s and the g_{o}s depend on the V_js. If $g_{\mathrm{o}i} \triangleq g_{\mathrm{on}i} + g_{\mathrm{op}i}$, then as defined in Section 2.2, for $\mathbf{y}, \mathbf{z} \in (0, V_{\mathrm{DD}})^n$,

$$\begin{aligned}
[K(\mathbf{y}, \mathbf{z})]_{ij} &= \Big[\int_0^1 g_{\mathrm{m}i}((1-\alpha)\mathbf{y} + \alpha\mathbf{z})\mathrm{d}\alpha\Big] x_{ij} \\
&\quad + \Big[\int_0^1 g_{\mathrm{o}i}((1-\alpha)\mathbf{y} + \alpha\mathbf{z})\mathrm{d}\alpha\Big] \\
&= \hat{g}_{\mathrm{m}i}(\mathbf{y}, \mathbf{z}) x_{ij} + \hat{g}_{\mathrm{o}i}(\mathbf{y}, \mathbf{z})
\end{aligned}$$

Since $\mathbf{y}, \mathbf{z} \in (0, V_{\mathrm{DD}})^n$ and $(0, V_{\mathrm{DD}})^n$ is a convex subset of $\mathbb{R}^n$, $g_{\mathrm{m}i}((1-\alpha)\mathbf{y} + \alpha\mathbf{z}) > 0$ and $g_{\mathrm{o}i}((1-\alpha)\mathbf{y} + \alpha\mathbf{z}) \geq 0$ for all $\alpha \in [0, 1]$. Therefore, $\hat{g}_{\mathrm{m}i}(\mathbf{y}, \mathbf{z}) > 0$ and $\hat{g}_{\mathrm{o}i}(\mathbf{y}, \mathbf{z}) \geq 0$. In other words, the diagonal matrix $\widehat{G}_{\mathrm{m}} = \mathrm{diag}(\hat{g}_{\mathrm{m}1}, \hat{g}_{\mathrm{m}2}, \ldots, \hat{g}_{\mathrm{m}n}) > 0$ and is hence invertible and the diagonal matrix $\widehat{G}_{\mathrm{o}} = \mathrm{diag}(\hat{g}_{\mathrm{o}1}, \hat{g}_{\mathrm{o}2}, \ldots, \hat{g}_{\mathrm{o}n}) \geq 0$. Clearly, the matrix K can be written as $K = \widehat{G}_{\mathrm{m}} X + \widehat{G}_{\mathrm{o}}$.

$$\begin{aligned}
\det(K) \neq 0 &\Leftrightarrow \det(\widehat{G}_{\mathrm{m}}) \det(X + \widehat{G}_{\mathrm{m}}^{-1} \widehat{G}_{\mathrm{o}}) \neq 0 \\
&\Leftrightarrow \det(X + D) \neq 0,
\end{aligned}$$

where the diagonal matrix $D = \widehat{G}_{\mathrm{m}}^{-1} \widehat{G}_{\mathrm{o}} \geq 0$. Hence, a sufficient condition for the POPL network of Figure 1.4 to have a unique operating point is that $\det(X + D) \neq 0$ for *all diagonal matrices* $D \geq 0$. In order to characterize the matrices with the above property, the following equivalence, which is given and proved as Theorem 5 in [42], is used:

Theorem 2.3.2 *If M is a real square matrix, then* $\det(M + D) \neq 0$ *for every diagonal matrix $D \geq 0$ if and only if M is a P_0-matrix and* $\det(M) \neq 0$.

Hence, the following can be concluded:

Theorem 2.3.3 *The operating point of the POPL network in Figure 1.4 is unique if the input connectivity matrix X satisfies the following conditions:*

1. $\det(X) \neq 0$*; i.e., X is invertible.*

2. *The principal minors of X are nonnegative; i.e., X is a P_0-matrix.*

If the ideal MITE expression has been used instead of the generic model, the necessary and sufficient condition for the operating point to be unique would have been just the first condition; i.e., $\det(X) \neq 0$. That this condition is not sufficient in practical circuits is shown by the following example:

Example 2.3.1 *Consider the POPL equations:*

$$\begin{aligned} I_{\text{o}1} &= I_{\text{i}1}^{-1}\, I_{\text{i}3}^{2} \\ I_{\text{o}2} &= I_{\text{i}2}^{-1}\, I_{\text{i}3}^{2} \end{aligned} \tag{2.17}$$

Two circuits that produce the above input-output equation according to the ideal MITE expression are shown in Figure 2.2(a) and Figure 2.2(d). In the terminology of [12], the matrix of powers is given by

$$\Lambda = \begin{bmatrix} -1 & 0 & 2 \\ 0 & -1 & 2 \end{bmatrix}$$

Ideally, both circuits are realizations of Equation (2.17) *since they satisfy* $\Lambda = Y_1 X_1^{-1} = Y_2 X_2^{-1}$ *[12], where* X_1 *and* Y_1 *are the input and output connectivity matrices for the circuit in Figure 2.2(a) and* X_2 *and* Y_2 *are the corresponding matrices for the circuit in Figure 2.2(d). Specifically,*

$$X_1 = \begin{bmatrix} 2 & 1 & 0 \\ 1 & 2 & 0 \\ 1 & 1 & 1 \end{bmatrix} \qquad Y_1 = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 2 \end{bmatrix}$$

and

$$X_2 = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 1 & 2 \\ 1 & 1 & 1 \end{bmatrix} \qquad Y_2 = \begin{bmatrix} 1 & 0 & 2 \\ 2 & 1 & 0 \end{bmatrix}$$

It is clear that all the principal minors of X_1 are positive while the principal minor $\left|\begin{smallmatrix} 1 & 2 \\ 1 & 1 \end{smallmatrix}\right|$ of X_2 is clearly negative. Hence, X_1 satisfies the conditions of Theorem 2.3.3 while X_2 does not. Both X_1 and X_2 are invertible and hence, the *ideal* condition for uniqueness (assuming the ideal MITE expression) is satisfied for both circuits. To determine via simulations if the circuits shown have a unique operating point or not, the transfer characteristic

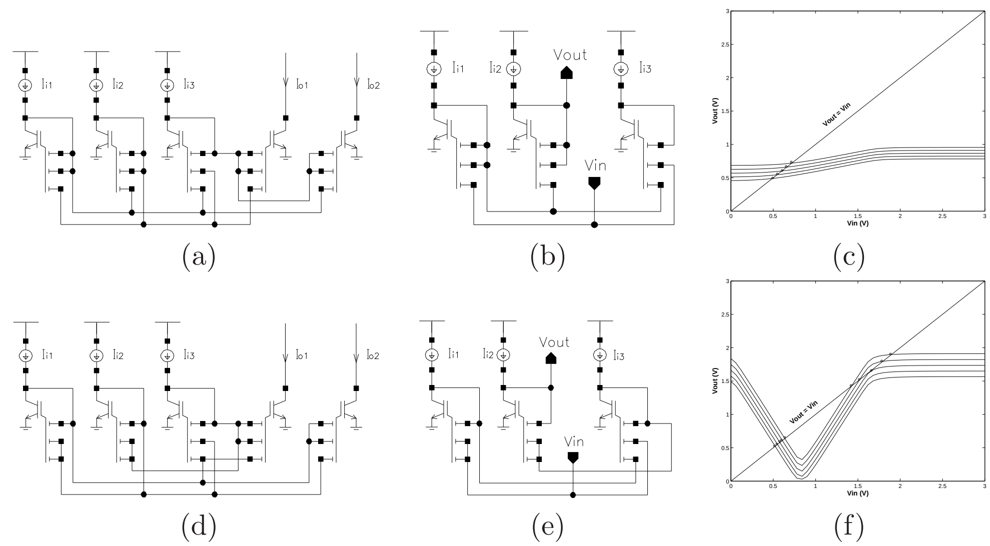


Figure 2.2. Example comparing the uniqueness criterion in Theorem 2.3.3 and the ideal uniqueness criterion. (a) Two circuits implementing $I_{\text{o}1} = I_{\text{i}1}^{-1} I_{\text{i}3}^{2}$; $I_{\text{o}2} = I_{\text{i}2}^{-1} I_{\text{i}3}^{2}$. (b) Circuits for finding the open-loop transfer characteristics. (c) Plots of the open-loop circuits for $I_{\text{i}2}$ varied logarithmically from $50nA$ to $500nA$.

(TC) method, alternatively called the positive feedback structure (PFBS) method described in [43,44,45,46], is used. The loop is broken at a convenient point and the open-loop transfer characteristic is calculated. If it is known that the open-loop circuit has a unique operating point, then the points of intersection of the open-loop transfer characteristic with the straight line of slope unity gives *all* the operating points of the closed-loop circuit. A way to break the loop in the circuits is shown in Figure 2.2(b) and Figure 2.2(e). That the open-loop circuits have a unique operating point (in the voltage range $(0, V_{\text{DD}})$) follows from a simple application of the fact that strictly monotonic functions from an interval of $\mathbb{R}$ into $\mathbb{R}$ are one-one.

A plot of the open-loop characteristics of the circuits for different values of $I_{\text{i}2}$ is shown in Figure 2.2(e) and Figure 2.2(f). It is clear that the circuit in Figure 2.2(a) has a unique operating point for all the chosen values of $I_{\text{i}2}$ while the other circuit has three operating points for some values of the current $I_{\text{i}2}$. Hence, it is clear that the uniqueness condition derived using the ideal MITE expression is not sufficient in practical circuits and that a more general condition like the one in Theorem 2.3.3 is required.

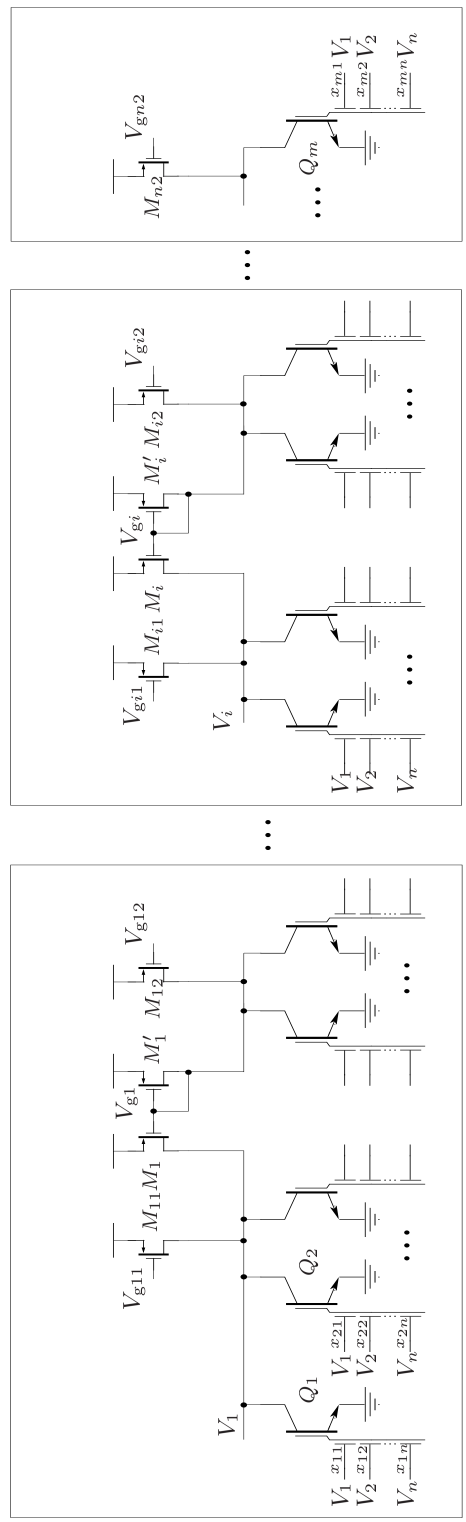


Figure 2.3. The dc circuit corresponding to a general MITE network described in Section 2.4.

2.4 The General MITE Network

The implementation of any (linear or nonlinear) ordinary differential equation as a MITE network will result in a set of equations either of the form $I_{\mathrm{c}i} = \sum_j I'_j - \sum_j I_j$ or $0 = \sum_j I'_j - \sum_j I_j$, where $I_{\mathrm{c}i}$ is the current through a capacitance and I_j, I'_j are (positive) currents that are generated from MITEs and/or PFET current sources. The dc circuit of this MITE network is obtained by setting $I_{\mathrm{c}i} = 0$ which results in the set of equations :

$$\sum_j I_{ij} = (\sum_j I'_{ij} - I_{\mathrm{b}i2}) + I_{\mathrm{b}i1} \quad i \in [1:n] \tag{2.18}$$

where the I_{b}s refer to the PFET current sources. The dc circuit of a general MITE network is shown in Figure 2.3. It has m MITEs that are split into n blocks, each block representing an equation in Equation (2.18). In the i^{th} block, the PFETs M_{i1} and M_{i2} generate the bias currents $I_{\mathrm{b}i1}$ and $I_{\mathrm{b}i2}$. The current mirror formed by the PFETs M_i and M'_i provide the current $(\sum_j I'_{ij} - I_{\mathrm{b}i2})$ in Equation (2.18).

The network topology is characterized by two matrices $A = [a_{ij}]$ and $X = [x_{ij}]$ defined by

$$\begin{aligned} a_{ij} &= \begin{cases} 1 & \text{if the drain of the MITE } Q_j \text{ is } V_i. \\ -1 & \text{if the drain of the MITE } Q_j \text{ is connected} \\ & \text{to } V_i \text{ through a current mirror.} \\ 0 & \text{otherwise} \end{cases} \\ x_{jk} &= \text{The weight through which } V_k \text{ is connected} \\ &\quad \text{to the MITE } Q_j. \end{aligned} \tag{2.19}$$

The MITE Q_j is said to be *attached* to the voltage V_i if $a_{ij} \neq 0$. By substituting the expressions for the drain current in the MITEs and the PFETs as given in Section 2.1, it is easy to see that the i^{th} block contributes two equations:

$$\sum_{\substack{j=1 \\ a_{ij}=1}}^{m} f_j\left(\sum_{k=1}^{n} x_{jk} V_k, V_i\right) = g'_i(V_{\mathrm{g}i}, V_i) + g_{i1}(V_{\mathrm{g}i1}, V_i)$$

$$g_i(V_{\mathrm{g}i}, V_{\mathrm{g}i}) + g_{i2}(V_{\mathrm{g}i2}, V_{\mathrm{g}i}) = \sum_{\substack{j=1 \\ a_{ij}=-1}}^{m} f_j\left(\sum_{k=1}^{n} x_{jk} V_k, V_{\mathrm{g}i}\right).$$

Hence, defining

$$
\begin{aligned}
F_i(V_1, V_{\mathrm{g}1}, \ldots, V_n, V_{\mathrm{g}n}) = & \sum_{\substack{j=1 \\ a_{ij}=1}}^{m} f_j\left(\sum\nolimits_{k=1}^{n} x_{jk} V_k, V_i\right) \\
& - g_i'(V_{\mathrm{g}i}, V_i) - g_{i1}(V_{\mathrm{g}i1}, V_i) \\
F_{\mathrm{g}i}(V_1, V_{\mathrm{g}1}, \ldots, V_n, V_{\mathrm{g}n}) = & \; g_i(V_{\mathrm{g}i}, V_{\mathrm{g}i}) + g_{i2}(V_{\mathrm{g}i2}, V_{\mathrm{g}i}) \\
& - \sum_{\substack{j=1 \\ a_{ij}=-1}}^{m} f_j\left(\sum\nolimits_{k=1}^{n} x_{jk} V_k, V_{\mathrm{g}i}\right),
\end{aligned}
\tag{2.20}
$$

the function $F : \mathbb{R}^{2n} \mapsto \mathbb{R}^{2n}$ (with the indices ordered as $1, \mathrm{g}1, 2, \mathrm{g}2, \ldots, n, \mathrm{g}n$) representing the $2n$-equations in the $2n$-variables is obtained. The elements of R_i and $R_{\mathrm{g}i}$ in the Jacobian matrix of F are given by

$$
\frac{\partial F_i}{\partial V} = \begin{cases}
\sum\limits_{a_{ij}=1}^{m} g_{\mathrm{m}j} x_{jk} & \text{if } V = V_k,\;\; k \neq i \\
\sum\limits_{a_{ij}=1}^{m} g_{\mathrm{m}j} X_{ji} + g_{\mathrm{o}i} & \text{if } V = V_i \\
0 & \text{if } V = V_{\mathrm{g}k},\;\; k \neq i \\
g_{\mathrm{m}\mathrm{g}i}' & \text{if } V = V_{\mathrm{g}i}
\end{cases}
\tag{2.21}
$$

$$
\frac{\partial F_{\mathrm{g}i}}{\partial V} = \begin{cases}
-\sum\limits_{a_{ij}=-1}^{m} g_{\mathrm{m}j} x_{jk} & \text{if } V = V_k \\
0 & \text{if } V = V_{\mathrm{g}k},\;\; k \neq i \\
-g_{\mathrm{m}\mathrm{g}i} - g_{\mathrm{o}\mathrm{g}i} & \text{if } V = V_{\mathrm{g}i},
\end{cases}
\tag{2.22}
$$

where $g_{\mathrm{o}i}$ and $g_{\mathrm{o}\mathrm{g}i}$ are the sum of the (nonnegative) output conductances of all the MITEs and PFETs connected to the nodes V_i and $V_{\mathrm{g}i}$, respectively, and hence are nonnegative. The functions $g_{\mathrm{m}j}$, $g_{\mathrm{m}\mathrm{g}i}'$, and $g_{\mathrm{m}\mathrm{g}i}$ are the transconductances of the MITE Q_j, the PFET M_i', and the PFET M_i, respectively, and are positive. Finding the K matrix, as defined in Theorem 2.2.1, for this function is equivalent to replacing each g_{m} and each g_{o} in Equation (2.21) and Equation (2.22) by the corresponding $\hat{g}_{\mathrm{m}}$ and $\hat{g}_{\mathrm{o}}$, like in the POPL network case. It should be noted that $C_{\mathrm{g}i}$ has only two nonzero entries, corresponding to R_i and $R_{\mathrm{g}i}$. A row transformation $R_i \mapsto R_i + \frac{\hat{g}_{\mathrm{m}\mathrm{g}i}'}{\hat{g}_{\mathrm{m}\mathrm{g}i} + \hat{g}_{\mathrm{o}\mathrm{g}i}} R_{\mathrm{g}i}$ results in a matrix with only one nonzero entry in $C_{\mathrm{g}i}$, which means that only a single lower-order determinant needs to be evaluated. Repeating the row transformation for each i, a $n \times n$ matrix K' is obtained whose elements are given by

$$[K']_{ik} = \sum_{j=1}^{m} a_{ij} b_j x_{jk} + d_i \delta_{ik},$$

where $d_i = \hat{g}_{\mathrm{o}i} \geq 0$ and $b_j > 0$ is either $\hat{g}_{\mathrm{m}j}$ or $\hat{g}_{\mathrm{m}j}\hat{g}'_{\mathrm{mg}i}/(\hat{g}_{\mathrm{mg}i} + \hat{g}_{\mathrm{og}i})$ depending on whether a_{ij} is nonnegative or negative. Thus, $K' = ABX + D$, where $B = \mathrm{diag}(b_1, b_2, \ldots, b_m)$ and $D = \mathrm{diag}(d_1, d_2, \ldots, d_m)$. Hence, the following theorem is proved:

Theorem 2.4.1 *The operating point of the network in Figure 2.3 is unique if the matrices* A *and* X *defined in Equation* (2.19) *are such that* $\det(ABX + D) \neq 0$ *for all diagonal matrices* $B > 0$ *and* $D \geq 0$*.*

2.5 Robust Criteria for Uniqueness of the Operating Point

The idea behind deriving the sufficiency conditions given in Theorems 2.3.3 and 2.4.1 is that one need not worry about the nonideality of the MITE(s) and depend only on the weights and the topology for deciding the uniqueness of the operating point. However, the weights themselves have been assumed to not vary. The weights are usually decided by certain capacitance ratios in practice and hence they can be assumed to be "quite" accurate, especially since they are usually integral multiples of an unit capacitance. However, one must still make sure that "very small" differences do not change a network satisfying the uniqueness conditions to one that does not. It should be noted that the uniqueness conditions reduce to checking whether some matrices have a nonnegative or positive determinant. It is clear from Bolzano's theorem that since the determinant function is multiaffine and hence a continuous function of its elements, a positive determinant implies that changing the elements by a sufficiently small amount still preserves the sign. Hence, one needs to check for robustness only for the case of matrices with a zero determinant. In this case, it can be shown that if all the elements of the matrix are allowed to be perturbed, then the determinant becomes negative no matter how small the perturbation. However, assuming that all the elements can be perturbed neglects the fact that elements of these matrices correspond to the weights connecting certain voltages in the network to MITEs. The errors in the input capacitances (which determine the weight) connecting a voltage to a MITE are the actual cause of this perturbation. In practice, only those voltages that have a nonzero weight connecting to a MITE are connected to the input capacitances. Though a zero weight is shown in the MITE symbol, in practice it is not connected through an input capacitance and hence the question of an error in this capacitance value does not arise. Hence, only the nonzero elements of the matrices under consideration need be perturbed. In this respect, the following theorem is proved:

Theorem 2.5.1 *If $M \in \mathcal{M}_n(\mathbb{R})$ and $\det(M) = 0$, then the determinant of M remains nonnegative under a perturbation of its nonzero elements if and only if each term in the standard determinant expansion of $\det(M)$ is zero, in which case the determinant of the perturbed matrix is also zero.*

The following lemma, proved in Appendix 2.A, is needed to prove Theorem 2.5.1:

Lemma 2.5.1 *If $f : \mathbb{R}^n \mapsto \mathbb{R}$ is such that*

1. *f is multiaffine i.e., for each variable x_i,*

   $f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) + x_i h(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$

2. *$f(\mathbf{0}) = 0$*

3. *There exists a $\delta > 0$ such that whenever $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ is such that $\|\mathbf{x}\|_\infty \triangleq \max(|x_1|, \ldots, |x_n|) < \delta$, then $f(\mathbf{x}) \geq 0$,*

then $f = 0$; i.e., $f(\mathbf{x}) = 0$ for all $\mathbf{x} \in \mathbb{R}^n$. In particular, the coefficient of $x_{i_1} x_{i_2} \cdots x_{i_k}$ $(k < n)$ in f is 0.

Proof A perturbation of the nonzero elements of $M = [m_{ij}]$ can be represented as $M'(\epsilon) = [m_{ij}(1 + \epsilon_{ij})]$; it should be noted that the zero elements are not perturbed in this case. The determinant of M' is given by the standard determinant expansion

$$
\begin{aligned}
f(\epsilon) \triangleq \det M' &= \sum_\sigma \operatorname{sign} \sigma \prod_{i=1}^{n} \left\{ m_{i\sigma(i)} (1 + \epsilon_{i\sigma(i)} \right\} \\
&= \sum_\sigma \operatorname{sign} \sigma \left\{ \prod_{i=1}^{n} m_{i\sigma(i)} \right\} \left\{ \prod_{i=1}^{n} (1 + \epsilon_{i\sigma(i)}) \right\},
\end{aligned}
\tag{2.23}
$$

where the sum runs over all $n!$ permutations σ of $[1 : n]$ and $\operatorname{sign} \sigma$ is 1 or -1 according to whether σ is an even or odd permutation, respectively. It is clear that f is multiaffine because the degree of ϵ_{ij} is at most 1 which follows from the determinant of a matrix being multiaffine with respect to the matrix elements. Also, $f(0) = \det M'(0) = \det M = 0$. Further, if it is assumed that perturbing the nonzero elements of M by a sufficiently small amount leaves its determinant nonnegative, then the assumption is essentially that $f(\epsilon) = \det M'(\epsilon)$ satisfies Condition (3) of Lemma 2.5.1. By Lemma 2.5.1, $f = 0$. By Equation (2.23), the coefficient of $\prod_{i=1}^{n} \epsilon_{i\sigma(i)}$ is $\prod_{i=1}^{n} m_{i\sigma(i)}$ which is 0 by the last conclusion of Lemma 2.5.1. Hence, each term in the standard determinant expansion of $\det M$, given by $\prod_{i=1}^{n} m_{i\sigma(i)}$, is 0 if $\det M$ is to remain nonnegative under an arbitrarily small perturbation of the nonzero elements of M. The converse statement is obvious from Equation (2.23). $\square$

Applying Theorem 2.5.1 to the criterion in Theorem 2.3.3, the following criterion is arrived at for a POPL network to have a (robust) unique operating point:

Theorem 2.5.2 *The operating point of a MITE POPL network is unique and remains unique under a sufficiently small perturbation of the MITE input capacitances if the input connectivity matrix X satisfies the following conditions:*

1. $\det(X) \neq 0$*; i.e., X is invertible.*

2. *The principal minors of X are nonnegative; i.e., X is a P_0-matrix.*

3. *If the principal minor corresponding to a submatrix X' is zero, then every term in the standard determinant expansion of X' is zero.*

A matrix satisfying conditions (2) and (3) of Theorem 2.5.2 will be called a "robust" P_0 matrix, abbreviated to RP_0-matrix. Specifically,

Definition 2.5.1 *A matrix $M \in \mathcal{M}_n(\mathbb{R})$ will be called a RP_0-matrix if:*

1. *M is a P_0-matrix.*

2. *If the principal minor corresponding to a submatrix M' is zero, then every term in the standard determinant expansion of M' is zero.*

The set of all $n \times n$ real RP_0-matrices will be referred to as $\mathcal{RP}0_n(\mathbb{R})$.

Hence, the set of matrices that satisfy the robust Theorem 2.5.2 is nothing but $\mathcal{GL}_n(\mathbb{R}) \cap \mathcal{RP}0_n(\mathbb{R})$, where $\mathcal{GL}_n(\mathbb{R})$ is the set of nonsingular real $n \times n$ matrices.

Similarly, a "robust" version of $\mathcal{P}0_n^+$, $\mathcal{RP}0_n^+$ can be defined by simply replacing P_0 in Definition 2.5.1 by P_0^+. It can be shown that $\mathcal{RP}0_n^+ = \mathcal{P}0_n^+ \cap \mathcal{RP}0_n$. Since all principal minors of a P-matrix are positive, a sufficiently small perturbation of any element, not necessarily the nonzero ones only, still preserves the sign of the principal minors and hence a "robust" version of $\mathcal{P}_n$ is $\mathcal{P}_n$ itself. The following inclusions clearly hold:

$$\mathcal{P}_n \subseteq \mathcal{RP}0_n^+ \subset \mathcal{RP}0_n \subseteq \mathcal{P}0_n \qquad \mathcal{P}_n \subseteq \mathcal{RP}0_n^+ \subseteq \mathcal{P}0_n^+ \subset \mathcal{P}0_n \tag{2.24}$$

Finally, a theorem about the relationship between $\mathcal{P}_n$ and $\mathcal{RP}0_n$ is presented below:

Theorem 2.5.3 *A $n \times n$ real matrix has a positive diagonal and is a RP_0-matrix if and only if it is a P-matrix. In other words, $\mathcal{RP}0_n \cap \{X \in \mathcal{M}_n(\mathbb{R}) \mid \operatorname{diag}(X) > 0\} = \mathcal{P}_n$.*

Proof If $X \in \mathcal{RP}0_n(\mathbb{R}) \cap \{X \in \mathcal{M}_n(\mathbb{R}) \mid \operatorname{diag}(X) > 0\}$, then all principal minors are nonnegative. Let $X' = [x'_{ij}]$ be a $k \times k$ principal submatrix ($k \leq n$) with zero determinant. By definition, every term in the standard determinant expansions of X' must be zero. Since $\operatorname{diag}(X) > 0$, it follows that $\operatorname{diag}(X') > 0$ as the diagonal elements are X' are also in the diagonal of X. Hence, there is at least one nonzero term in the determinant expansion of X', given by $x'_{11}x'_{22}\cdots x'_{kk}$. This contradicts the assumption that $\det(X') = 0$. Hence, all the principal minors of X are positive; i.e., $X \in \mathcal{P}_n$.

Conversely, if $X = [x_{ij}] \in \mathcal{P}_n$, $\operatorname{diag}(X) > 0$ since each diagonal element x_{ii} of X is a principal minor ($x_{ii} = X(\{i\})$). Also, $\mathcal{P}_n \subset \mathcal{RP}0_n$ from the inclusion relations in Equation (2.24). Hence, $\mathcal{P}_n \subseteq \mathcal{RP}_n \cap \{X \in \mathcal{M}_n(\mathbb{R}) \mid \operatorname{diag}(X) > 0\}$. $\square$

Theorem 2.5.3 tells us that if we assume that the input connectivity matrix X has a positive diagonal, then the only way that a POPL network has a unique operating point that remains insensitive to floating-gate capacitor mismatch is when X is a P-matrix. It should be noted that being a P-matrix is stronger than being a RP_0-matrix or a P_0-matrix. Further, requiring all the principal minors of X to be positive makes the unique operating point property insensitive to a sufficiently small change in any element of X, irrespective of whether that element is zero or nonzero.

2.6 Stability of the Operating Point of POPL Networks

MITE implementations require a multiple-input voltage summer. In the absence of good large resistors occupying small area in current technologies, the use of capacitors for this voltage summation is inevitable. The use of these input capacitors increases the order of the system and hence the stability of the system becomes an issue. As shown in Chapter 1, an analysis of the stability of the POPL networks exists in literature; however, it is limited by the fact that that it does not take into account the input capacitors as the whole input capacitor network is replaced by parasitic capacitors from each node to ground. Also, the capacitors are included only in the input MITEs; the output MITEs are not taken into account for stability considerations in this analysis. In this section, we derive conditions for stability that take these effects into account; but first, we mentions some known results on D-stability.

There are no known finitely verifiable necessary and sufficient conditions for D-stability for order greater than 3. Several sufficient conditions are given in [47]. The following theorem, proved in [39], states an important necessary condition:

Theorem 2.6.1 *A D-stable matrix is a P_0^+-matrix; i.e., $\mathcal{D}_n(\mathbb{R}) \subseteq \mathcal{P}0_n^+(\mathbb{R})$*

From the definitions of different types of matrices in Section 2.2.0.3, the following inclusions are easily observed:

$$\mathcal{D}_n \subseteq \mathcal{P}0_n^+ \subset \mathcal{P}0_n \qquad\qquad \mathcal{P}_n \subseteq \mathcal{P}0_n^+ \subset \mathcal{P}0_n \tag{2.25}$$

Equality in the above inclusions is not valid in general except for $\mathcal{D}_{1,2} = \mathcal{P}0_{1,2}^+$. For $n = 3$, the following characterization of $\mathcal{D}_n$ exists [48] (slightly rephrased from the original version):

Lemma 2.6.1 *Let $A = [a_{ij}]$ be a real 3×3 matrix. Let m_i be the cofactor of a_{ii} for $i = 1, 2, 3$. Let $\Delta = \left(\sum_{i=1}^{3} \sqrt{(}a_i m_i)\right)^2 - \det(A)$. Then, A is D-stable if and only if*

1. *A is a P_0^+ matrix, which in this case reduces to a_{11}, a_{22}, a_{33}, m_1, m_2, m_3 being non-negative and $a_{11} + a_{22} + a_{33}$, $m_1 + m_2 + m_3$, and $\det(A)$ being positive.*

2. *$\Delta \geq 0$*

3. *If $\Delta = 0$, then for some $i \in \{1, 2, 3\}$, $a_{ii} m_i$ is zero with one of a_{ii}, m_i being nonzero.*

A useful sufficient for D-stability is diagonal stability.

Definition 2.6.1 *A matrix $M \in \mathcal{M}_n(\mathbb{R})$ is said to be diagonally stable if it has a positive diagonal Lyapunov solution i.e., there exists a diagonal matrix $P > 0$ such that $PM + M^T P$ is positive definite.*

While there is no "finite characterization" for diagonal stability too, there is a numerical procedure for doing so because to test if there is a positive diagonal matrix $P > 0$ such that $PM + M^T P$ is positive definite is to check the feasibility of a *linear matrix inequality* (LMI) and there are polynomial-time algorithms for solving this (for example, through MATLAB's LMI Control Toolbox). A MATLAB code for testing diagonal stability using this toolbox is given in [49].

2.6.1 New stability criterion for POPL networks

The D-stability test was arrived at using the assumption that the only significant capacitances of interest are capacitances from each node to ground. We will consider a different model which has the D-stability criterion as a limiting case, but in general takes into account both the physical floating-gate capacitances, variously called as the "input capacitances" or the "control-gate capacitances", as well as the parasitic capacitances in the floating-gate MOSFET itself. For this, we will analyze a POPL network by linearizing the circuit around some operating point, which is clearly equivalent to doing a small-signal ac analysis to find the characteristic equation.

The small-signal model we are assuming is based upon the derivation of the model of a subthreshold floating-gate MOSFET given in [12] and mentioned in Chapter 1. In case a cascode transistor is present, the drain of the floating-gate MOSFET is largely fixed. Hence, the error is small in assuming that any capacitance present between the floating-gate and the drain is simply a capacitance between the floating-gate and ground. The model is shown in Figure 2.4. Here, C_{p} is equal to $(C_{\mathrm{ox}}C_{\mathrm{dep}})/(C_{\mathrm{ox}} + C_{\mathrm{dep}}) + C_{\mathrm{b}} + C_{\text{fg-s}} + C_{\text{fg-d}}$. The output resistance of the MITE is not used in the small-signal model here mainly because the input MITEs are diode-connected by a nonzero weight, i.e., $x_{ii} > 0$ is an usual assumption in the synthesis. If $X \in \mathcal{M}_n(\mathbb{R})$ and $Y \in \mathcal{M}_{l\times n}(\mathbb{R})$ are the input and output connectivity matrices, the matrix $Z = \left[\begin{smallmatrix} X \\ Y \end{smallmatrix}\right]$ will be called the *connectivity matrix*. We define $m = l+n$. To

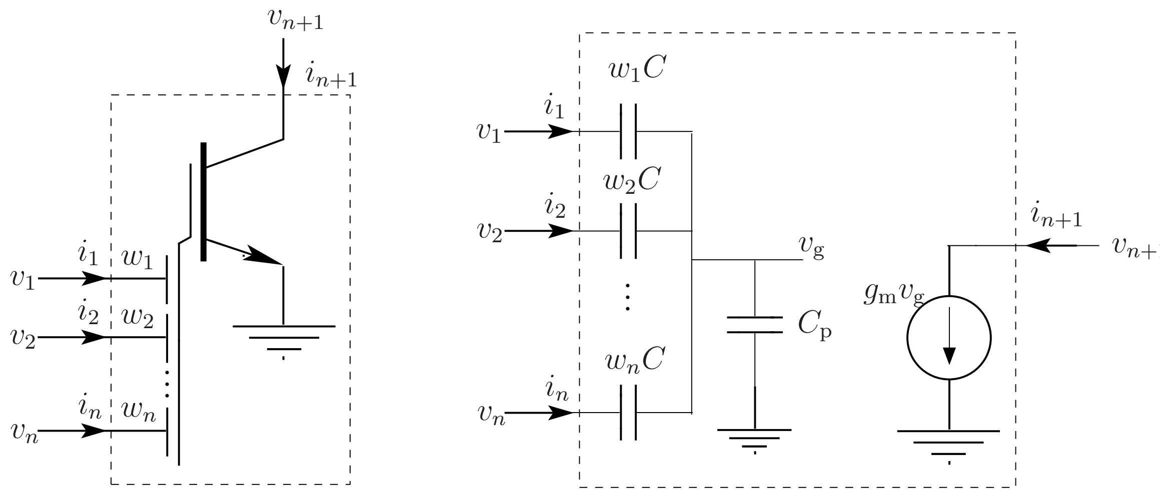


Figure 2.4. The small-signal equivalent model used for analyzing the stability properties of a POPL MITE network. v_g is the small-signal floating-gate voltage. C is the unit capacitance that the floating-gate capacitances are made of. C_p is the sum of the capacitances from the floating-gate to bulk, source, and drain. As a first-order approximation, the other capacitances in the network are neglected.

find the characteristic equation, we will determine the *nodal admittance matrix* [50] of the circuit, which is a standard procedure. It should be noted that for stability considerations, the controlled source in a output MITE does not enter the equations; hence we number the nodes starting with the drains of the input MITEs (size n) followed by the floating-gate voltages of the MITEs.

The nodal admittance matrix $Y(s)$ is then found out to be

$$Y(s) = \begin{bmatrix} sC\,\mathrm{diag}(Z^T\mathbf{1}_m) & G - sCZ^T \\ -sCZ & sC\,\mathrm{diag}(Z\mathbf{1}_n) + sC_p I_m \end{bmatrix} \tag{2.26}$$

Here $G = [g_m\ 0_{n\times l}]$, where g_m is the $n \times n$ diagonal matrix whose ii^{th} element is g_{mi}, the g_m of the i^{th} input MITE. Also, when $\mathbf{v}$ is a vector, $\mathrm{diag}(\mathbf{v})$ refers to the square matrix with $\mathbf{v}$ as the diagonal. In order that the POPL network be stable, we want the characteristic equation $\det(Y(s)) = 0$ to have roots in the open left–half s–plane. Note that the presence of the cutset of capacitors in the model leads to presence of m roots for the characteristic equation at the origin. The remaining roots, however, need to be in the open left–half s–plane.

Some simplifications can be made to the characteristic equation. Assuming that the POPL network is balanced, it follows that $\mathrm{diag}(Z\mathbf{1}_n) = WI_m$, where W is the fan–in of the network. The determinant of a matrix can be represented as the product of the determinant of a principal submatrix and the determinant of its Schur complement [28]. Hence, we have

$$
\begin{aligned}
\det(Y(s)) = {} & \det(sC\,\mathrm{diag}(Z\mathbf{1}_n) + sC_{\mathrm{p}} I_m) \\
& \det\big(sC\,\mathrm{diag}(Z^T\mathbf{1}_m) - (G - sCZ^T)(sC\,\mathrm{diag}(Z\mathbf{1}_n) + sC_{\mathrm{p}} I_m)^{-1}(-sCZ)\big) \\
= {} & s^m C^m (W + \frac{C_{\mathrm{p}}}{C})^m \det\big(sC(\mathrm{diag}(Z^T\mathbf{1}_m) - \frac{Z^T Z}{W + C_{\mathrm{p}}/C}) + \frac{[g_{\mathrm{m}}\ 0_{n\times l}]}{W + C_{\mathrm{p}}/C}\begin{bmatrix} X \\ Y \end{bmatrix}\big) \\
= {} & s^m C^m (W + \frac{C_{\mathrm{p}}}{C})^m \det\big(sC(\mathrm{diag}(Z^T\mathbf{1}_m) - \frac{\epsilon}{W} Z^T Z) + \frac{g_{\mathrm{m}} X}{W + C_{\mathrm{p}}/C}\big) \\
= {} & s^m C^m (W + \frac{C_{\mathrm{p}}}{C})^m \det(X)/W^n \det\big(sC(W\,\mathrm{diag}(Z^T\mathbf{1}_m) - \epsilon Z^T Z)X^{-1} + D\big)
\end{aligned}
\tag{2.27}
$$

Here $\epsilon \triangleq W/(W + C_{\mathrm{p}}/C)$ clearly lies between 0 and 1 , and $D = \frac{g_{\mathrm{m}}}{W + C_{\mathrm{p}}/C}$ is clearly a diagonal matrix with a positive diagonal. We have therefore proved the following result:

Theorem 2.6.2 *The POPL network described by the connectivity matrix Z is stable for all values of input currents if and only if $F(\epsilon) = (W\,\mathrm{diag}(Z^T\mathbf{1}_m) - \epsilon Z^T Z)X^{-1}$ is a D-stable matrix, where $\epsilon = W/(W + C_{\mathrm{p}}/C)$ and $C_{\mathrm{p}} = (C_{\mathrm{ox}} C_{\mathrm{dep}})/(C_{\mathrm{ox}} + C_{\mathrm{dep}}) + C_{\mathrm{b}} + C_{\mathrm{fg-s}} + C_{\mathrm{fg-d}}$.*

It can be observed that the condition described in Theorem 2.6.2 differs from the stability condition in Theorem 1.3.2 in the following ways:

1. The output connectivity matrix enters the stability criterion in Theorem 2.6.2 through Z but is absent in Theorem 1.3.2. Clearly, this is a result of neglecting the "loading" of the output MITEs in Theorem 1.3.2.

2. The condition in Theorem 2.6.2 depends upon the value of the parasitic capacitance from the floating gate to ground. As ϵ tends to 0, we find that the network is stable if X^{-1} and consequently, X, is D-stable. Hence, the earlier condition in Theorem 1.3.2 is a limiting case of Theorem 2.6.2 as ϵ tends to zero or equivalently, as $C_{\mathrm{p}} \longrightarrow \infty$.

As mentioned before, F is D-stable if it is diagonally stable. When the exact value of ϵ is not known, it is desirable to verify the diagonal stability of F when ϵ belongs to the set $[\epsilon_1, \epsilon_2]$. Here, the fact that F is linear in ϵ can be used to obtain the following sufficient condition:

Lemma 2.6.2 *$F(\epsilon)$ is D-stable for all $\epsilon \in [\epsilon_1, \epsilon_2]$ if $F(\epsilon_1)$ and $F(\epsilon_2)$ are simultaneously diagonally stable i.e., there is a diagonal matrix $P > 0$ such that both $F(\epsilon_1)P + PF(\epsilon_1)^T$ and $F(\epsilon_2)P + PF(\epsilon_2)^T$ are positive definite.*

It should be noted that verifying the above condition reduces to checking the feasibility of a LMI.

2.7 Appendix 2.A

In this appendix, Lemma 2.5.1 is proved. A function $f : \mathbb{R} \mapsto \mathbb{R}$ is said to be affine if for all $x \in \mathbb{R}$, $f(x) = a + bx$ (for some constants $a, b \in \mathbb{R}$). A function $f : \mathbb{R}^n \mapsto \mathbb{R}$ is said to be multiaffine if it is affine in each variable; i.e., for each variable x_i,

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n) + x_i h(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$$

A equivalent definition would be to define a multiaffine function as a polynomial in which every variable has degree at most 1.

Claim 2.7.1 *If $f : \mathbb{R}^n \mapsto \mathbb{R}$ is such that*

1. *f is multiaffine*
2. *$f(\mathbf{0}) = 0$*
3. *There exists a $\delta > 0$ such that whenever $\mathbf{x} = (x_1, \ldots, x_n) \in \mathbb{R}^n$ is such that $\|\mathbf{x}\|_\infty \triangleq \max(|x_1|, \ldots, |x_n|) < \delta$, then $f(\mathbf{x}) \geq 0$,*

then $f = 0$. In particular, the coefficient of $x_{i_1} x_{i_2} \cdots x_{i_k}$ $(k \leq n)$ in f is 0.

Proof The lemma is proved by induction on n. When $n = 1$, $f(x_1) = a + bx_1$ for some $a, b \in \mathbb{R}$. $f(\mathbf{0}) = 0$ implies $a = 0$ and hence $f(x_1) = bx_1$. When $|x_1| < \delta$, $f(x_1) \geq 0$. However, if $b \neq 0$, then by choosing $x_1 = -\operatorname{sign}(b)\delta/2$, $f(x_1) = -|b|\delta/2 < 0$ is obtained, which contradicts the requirement that $f(x_1) \geq 0$. Hence, $b = 0$ and the basis for induction is proved.

By the multiaffinity of f, there exist functions $g, h : \mathbb{R}^{n-1} \mapsto \mathbb{R}$ such that

$$f(x_1, \ldots, x_n) = g(x_1, \ldots, x_{n-1}) + x_n h(x_1, \ldots, x_{n-1})$$

It will be shown that g and h satisfy the constraints in the lemma so that by induction, $g = h = 0$ and hence $f = 0$. First, consider $g(x_1, \ldots, x_{n-1})$, which by the above expression is given by $f(x_1, \ldots, x_{n-1}, 0)$. Clearly, g is multiaffine since f is. Also, $g(0, \ldots, 0) = f(0, \ldots, 0) = 0$. Further, let $\delta > 0$ be such that $\|\mathbf{x}\|_\infty < \delta$ implies $f(\mathbf{x}) \geq 0$. If $< x_1, \ldots, x_{n-1} >$ is such that $\max(|x_1|, \ldots, |x_{n-1}|) < \delta$, then it is clear that $\max(|x_1|, \ldots, |x_{n-1}|, 0) < \delta$, which shows that $g(x_1, \ldots, x_{n-1}) = f(x_1, \ldots, x_{n-1}, 0) \geq 0$. By induction, $g = 0$. Hence,

$$f(x_1, \ldots, x_n) = x_n h(x_1, \ldots, x_{n-1}) \tag{2.28}$$

Clearly, h is multiaffine. Choose $< x_1, \ldots, x_{n-1} >$ such that $\max(|x_1|, \ldots, |x_{n-1}|) < \delta$. Clearly, both $x_n = \delta/2$ and $x_n = -\delta/2$ satisfy $\max(|x_1|, \ldots, |x_{n-1}|, |x_n|) < \delta$ and hence, from Equation (2.28),

$$\begin{aligned} \frac{\delta}{2}\, h(x_1, \ldots, x_{n-1}) &\geq 0 \\ -\frac{\delta}{2}\, h(x_1, \ldots, x_{n-1}) &\geq 0 \end{aligned}$$

Since $\delta > 0$, the above implies that for all $< x_1, \ldots, x_{n-1} >$ such that $\max(|x_1|, \ldots, |x_{n-1}|) < \delta$, $h(x_1, \ldots, x_{n-1}) = 0$, which satisfies Condition (3) of the lemma trivially. This also shows that $h(0, \ldots, 0) = 0$, which is simply Condition (2). By induction, $h = 0$. Hence, $f = 0$. The coefficient of $x_{i_1} x_{i_2} \cdots x_{i_k}$ $(k \leq n)$ in f is 0 if any of the indices $i_1, i_2, \ldots, i_k$ are equal. This is because the degree of each variable is at most 1 in a multiaffine function. If the indices are distinct, then the coefficient is simply $\frac{\partial^k f}{\partial x_{i_1} \partial x_{i_2} \cdots \partial x_{i_k}}(0)$ which is 0 since f is the zero function. $\square$

CHAPTER 3
SYNTHESIS OF MITE TRANSLINEAR LOOPS

3.1 Translinear Loops

It was observed in the section on translinear circuits that translinear loops form a important part of the implementation of equations using translinear circuits. The implementation of a system of translinear loop equations using MITE circuits is now discussed [17].

A *system of translinear–loop equations* (STLE) is defined as a relationship between current variables $I_1, I_2, \ldots, I_m$ of the form

$$\begin{array}{cccc} I_1^{a_{11}} & I_2^{a_{12}} & \cdots & I_m^{a_{1m}} = 1 \\ I_1^{a_{21}} & I_2^{a_{22}} & \cdots & I_m^{a_{2m}} = 1 \\ & \vdots & & \vdots \\ I_1^{a_{l1}} & I_2^{a_{l2}} & \cdots & I_m^{a_{lm}} = 1 \end{array} \tag{3.1}$$

The matrix $A = [a_{ij}]$ represents the powers to which the currents are raised and will be referred to as the *translinear loop matrix*. Since the powers of interest usually are rational numbers, it follows that without loss of generality, $A \in \mathcal{M}_{l,m}(\mathbb{Z})$ can be assumed. Dimensional consistency requires that

$$\sum_{j=1}^{m} a_{ij} = 0 \qquad i \in [1:l],$$

which can be written in a more compact manner as

$$A\mathbf{1}_m = 0 \tag{3.2}$$

Taking logarithms on both sides of Equation (3.1),

$$A \log(\mathbf{I}) = 0 \tag{3.3}$$

It is clear that for purposes of synthesis, it can be assumed that the rows of A are linearly independent. This is just another way of stating that there are no redundant equations in Equation (3.1). Hence, the following is assumed:

Convention 3.1.1 *If A is a translinear loop matrix, then A is full-row-rank; i.e.,* rank $A = l$.

3.1.0.1 Input-Output Separation

Since $\mathrm{rank}(A) = l$, l linearly independent columns of A, indexed by γ, can be chosen. If $\beta = [1 : l]$, then the matrix $A(\beta, \gamma)$ is a nonsingular square matrix. Equation (3.3) can thus be written as $A(\beta, \gamma)\log(\mathbf{I}(\gamma)) + A(\beta, \gamma')\log(\mathbf{I}(\gamma')) = 0$, where by definition, the vector $\mathbf{I}(\gamma')$ represents the currents in $\mathbf{I}$ indexed by the indices *not* in γ. Thus,

$$\log(\mathbf{I}(\gamma)) = -A(\beta, \gamma)^{-1}A(\beta, \gamma')\log(\mathbf{I}(\gamma')). \tag{3.4}$$

This means that the n currents in $\mathbf{I}(\gamma')$ can be taken to be inputs and the l currents in $\mathbf{I}(\gamma)$ to be outputs to the MITE network. This formulation is nothing but the POPL formulation of Chapter 1. It is also clear that the POPL relationship can be written as a STLE by simply dividing each equation in Equation (1.6) by the corresponding output current. The difference between the two formulations is that in a STLE, the input and output currents are not explicitly separated. However, since all the criteria related to the uniqueness of the operating point and the stablility of the POPL network explicitly require a separation of the current signals into input and output currents, any synthesis taking into account these criteria should also take this difference into account. For this reason, the following convention will be followed with respect to system of translinear loops:

Convention 3.1.2 *The currents in the STLE are numbered so that the currents meant to be inputs to the MITE network have lower indices than the currents meant to be outputs. In other words, the currents $I_1, I_2, \ldots, I_n$ are the inputs to the system and the currents $I_{n+1}, I_{n+2}, \ldots, I_m$ are the outputs, where $n = m - l$. For this assumption to be valid, the matrix $A(\beta, \gamma)$ must be nonsingular, where $\gamma = [n + 1 : m]$ and $\beta = [1 : l]$.*

The purpose of this chapter is to describe the synthesis of MITE networks so that given the matrix A, the currents through the MITEs satisfy the relation in Equation (3.3). The resultant MITE network is optimal in a certain sense that will be described. This will be followed by the synthesis subject to the constraints in Chapter 2. It is shown that the synthesis of MITE networks is connected to the study of linear diophantine equations. This connection is explored and the results pertaining to the field of diophantine equations is used in the synthesis.

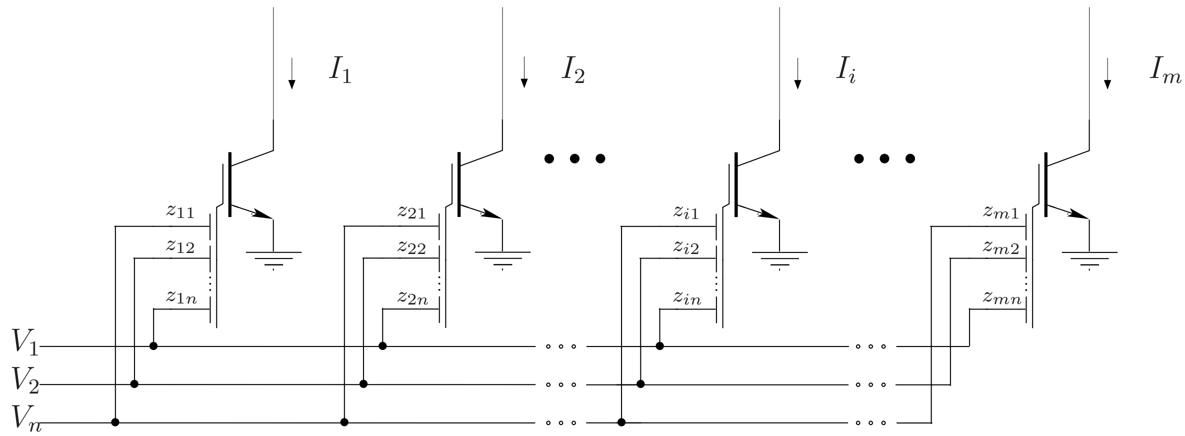


Figure 3.1. The canonical MITE network used to implement STLE Equation (3.1)**. The voltages $V_1, V_2, \dots, V_n$ are generated by "diode" connecting them to the respective drains of the input MITEs with currents $I_1, I_2, \dots, I_n$.**

3.2 Reformulation of POPL Networks

Consider the MITE network in Figure. 3.1. The matrix $Z = [z_{ij}] \in \mathcal{M}_{m,n}(\mathbb{N})$, called the *connectivity matrix*, represents the nonnegative integer weight coefficients connecting the voltages $\mathbf{V} = [V_i] \in \mathbb{R}^n$ to the MITEs. By definition, $\log\left\{\frac{I_i}{I_\mathrm{s}}\right\} = \frac{\kappa}{U_\mathrm{T}} \sum_{j=1}^{n} z_{ij} V_j$, which can be written as

$$Z\mathbf{V} = \frac{U_\mathrm{T}}{\kappa} \log\left\{\frac{\mathbf{I}}{I_\mathrm{s}}\right\} \tag{3.5}$$

In practice, the voltages $\mathbf{V}$ will be generated from the circuit itself by means of drain connections. The circuit thus obtained is clearly no different from a POPL network. Here, the Z matrix is simply a way to represent the *input and output connectivity matrices* of POPL networks as a single matrix. For instance, if the currents are ordered so that the first n are inputs and the next $l = m - n$ currents are outputs, and if X and Y are the input and output connectivity matrices, respectively, then the connectivity matrix of this network is clearly $Z = \left[\begin{smallmatrix} X \\ Y \end{smallmatrix}\right]$; i.e., the first n rows of Z form X and the next l rows form Y.

3.3 The Synthesis Problem

The synthesis problem is the reverse problem of the analysis presented in the last section; the objective is to find a suitable connectivity matrix $Z \in \mathcal{M}_{m,n}$ when the translinear loop matrix $A \in \mathcal{M}_{l,m}(\mathbb{Z})$ is given. In other words, it is desired that the set $\{I_\mathrm{s} \exp(\kappa \mathbf{U}/U_\mathrm{T}) | \mathbf{U} = Z\mathbf{V}$ for some $\mathbf{V} \in \mathbb{R}^n\}$, representing the set-theoretic *relation* determined by Z, be the same as the set $\{\mathbf{I} \in \mathbb{R}^m \mid A\log(\mathbf{I}) = 0\}$, which represents the STLE. Consider the vector $\mathbf{U} = [U_i] \in \mathbb{R}^m$ defined by

$$U_i \triangleq \frac{U_{\mathrm{T}}}{\kappa}\log(\frac{I_i}{I_{\mathrm{s}}}) = \frac{U_{\mathrm{T}}}{\kappa}\log(I_i) - \frac{U_{\mathrm{T}}}{\kappa}\log(I_{\mathrm{s}}) \tag{3.6}$$

Therefore, $\mathbf{U} = \frac{U_{\mathrm{T}}}{\kappa}\log(\mathbf{I}) - \frac{U_{\mathrm{T}}}{\kappa}\log(I_{\mathrm{s}})\mathbf{1}_m$. Clearly, $\mathbf{I}$ satisfies $A\log(\mathbf{I}) = 0$ iff

$$A\mathbf{U} = \frac{U_{\mathrm{T}}}{\kappa}A\log(\mathbf{I}) - \frac{U_{\mathrm{T}}}{\kappa}\log(I_{\mathrm{s}})A\mathbf{1}_m = 0 \tag{3.7}$$

The desired set equality can now be expressed as the requirement that $\{\mathbf{U}|\mathbf{U} = Z\mathbf{V}$ for some $\mathbf{V} \in \mathbb{R}^n\} = \{\mathbf{U} \in \mathbb{R}^m \mid A\mathbf{U} = 0\}$. The former is the range, $\mathrm{Im}(Z)$, of Z and the latter is the kernel, $\ker(A)$, of A. Hence, $\mathrm{Im}(Z) = \ker(A)$ is desired. The following result gives an equivalent characterization of this requirement.

Claim 3.3.1 $\mathrm{Im}(Z) = \ker(A)$ *if and only if* $AZ = 0$ *and* $\mathrm{rank}(Z) = \mathrm{nullity}(A)$.

Proof : If $\mathrm{Im}(Z) = \ker(A)$, then for any $\mathbf{V} \in \mathbb{R}^n$, $(AZ)\mathbf{V} = A(Z\mathbf{V}) = 0$. Thus the *linear transformation* $AZ = 0$, which means that the *matrix* AZ is 0. Clearly, $\mathrm{Im}(Z) = \ker(A)$ implies that the dimensions of these sets are also equal, which means $\mathrm{rank}(Z) = \mathrm{nullity}(A)$. Conversely, $AZ = 0$ means that an arbitrary element $Z\mathbf{V}$ of $\mathrm{Im}(Z)$ satisfies $A(Z\mathbf{V}) = (AZ)(\mathbf{V}) = 0$; i.e., $\mathrm{Im}(Z) \subseteq \ker(A)$. Since the dimensions of these two sets are equal, $\mathrm{Im}(Z)$ cannot be a proper subspace of $\mathrm{nullity}(A)$; i.e., $\mathrm{Im}(Z) = \ker(A)$. $\square$

If Conventions 3.1.1 and 3.1.2 are taken into account, then the $\mathrm{rank}(Z) = \mathrm{nullity}(A) = n$ requirement can be shown to reduce to Z being of the form $\left[\begin{smallmatrix} X \\ Y \end{smallmatrix}\right]$, where $X \in \mathcal{M}_{n,n}(\mathbb{N})$ is nonsingular. Taking into account these constraints as well as the ones in [12, 51] described in Chapter 2, the synthesis problem can be restated as

Given $A \in \mathcal{M}_{l,m}(\mathbb{Z})$. If $\gamma = \{n+1, n+2, \ldots, m\}$ and $\beta = [1:l]$, $A(\beta, \gamma)$ is nonsingular.

Problem Find a matrix $Z \in \mathcal{M}_{m,n}(\mathbb{N})$ satisfying:

P1 $AZ = 0$.

P2 $Z\mathbf{1}_n = w\mathbf{1}_m$ for some $w \in \mathbb{N}$. This ensures that the MITE network is balanced [19,17].

P3 If $\alpha = [1:n]$, then $X = Z(\alpha, \alpha)$ is nonsingular. This implies that $\mathrm{rank}(Z) = \mathrm{nullity}(A)$.

P4 X is a RP_0 matrix; i.e., $X \in \mathcal{RP}0_n(\mathbb{R})$. This ensures that the operating point of the MITE network is unique and that it is not affected by perturbations in the floating–gate capacitance values.

P5 X is D-stable; i.e., the eigenvalues of DX lie in the right–half s–plane for all diagonal matrices D with a positive diagonal. This implies that the MITE network is stable in the sense described in [12].

Conditions P4 and P5 assume that voltage V_i is connected to the drain of the MITE with current I_i, for $i \in [1 : n]$.

Some of the important parameters that need to be minimized are the number of MITEs and the fan–in of each MITE. Increasing either of these parameters usually results in a increase in chip area. For the same floating–gate capacitance value, if the fan–in is increased, the maximum frequency of operation of the circuit decreases. The synthesis methods in [12,18,19] are mainly for implementing each equation in the STLE separately. Once a MITE network is found for each equation, consolidation is used to remove redundant MITEs based on identifying voltages with the same value from different MITE networks. If consolidation is not possible for all voltages, then the final network has copies of the input currents flowing through different MITEs and hence the procedure is not optimal with respect to the number of MITEs. On the other hand, these methods can potentially reduce the fan–in, and it follows from [19, 12] that the fan–in can be reduced to the minimum possible value of 2. However, there is no procedure to minimize the number of MITEs once the fan–in is fixed at some value.

The optimal synthesis procedure presented here aims at synthesizing the STLEs as a whole rather than synthesizing each equation separately. The synthesis procedure is optimal in the following sense:

1. The minimum number of MITEs required for implementing Equation (3.1), viz. m, is attained.

2. The minimum fan–in is obtained amongst all MITE networks *with m MITEs* implementing the translinear–loop equations.

3.4 Operating Point Uniqueness and Stability

Condition P4 and P3 ensure that the POPL network has a unique operating point. In view of Theorem 2.5.3, if Z or X is also required to have a positive diagonal, then it is equivalent to assuming that X is a P-matrix. Hence, for synthesis purposes, instead of P4, the condition P4´ below shall be used:

P4´ X is a P-matrix; i.e., all principal minors of X are positive.

While directly testing all the principal minors for being nonnegative is, in general, of order $\mathrm{O}(n^3 2^n)$, an algorithm of order $\mathrm{O}(2^n)$ for testing if a matrix is a P-matrix or not has been proposed [52] which is used in the synthesis procedure. By Theorem 2.6.1, the condition P5 is actually a sufficient condition for P4 and P3 but does not imply P4´; i.e., that X is a P-matrix. However, no finitely verifiable necessary and sufficient condition exists for checking D-stability [39], though there are useful sufficient conditions [47]. Hence, the synthesis algorithm to be proposed is incomplete in the sense that it might result in MITE networks for which we cannot test for stability, if it cannot be tested by the available conditions.

3.5 Solution Methodology

The solution(s) of the synthesis problem taking into account conditions P1, P2, and P3 is first discussed.

If Z is written in terms of its columns; i.e., $Z = [\mathbf{z}_1 \mathbf{z}_2 \ldots \mathbf{z}_n]$, then $AZ = [A\mathbf{z}_1 A\mathbf{z}_2 \ldots A\mathbf{z}_n]$. Then the problem (P1, P2, and P3) is equivalent to finding a set $\{\mathbf{z}_i\}_{i=1}^n$ with $\mathbf{z}_i \in \mathbb{N}^m$ so that the following are satisfied:

R1 $A\mathbf{z}_i = 0 \qquad i \in [1:n]$

R2 $\Sigma_{i=1}^n \mathbf{z}_i \ = \ w\mathbf{1}_m$

R3 The vectors $\{\mathbf{z}_i\}$ are linearly independent. This is equivalent to the vectors $\{\mathbf{x}_i\}$ being linearly independent, where $\mathbf{x}_i = \mathbf{z}_i([1:n])$.

Some observations based on the conditions specified so far now follow.

Theorem 3.5.1 *If $Z = [\mathbf{z}_1\, \mathbf{z}_2 \cdots \mathbf{z}_n]$ satisfies P1, P3, P4/P4´, and P5, where the elements of Z are nonnegative real numbers, then so does $Z' = [\alpha_1\mathbf{z}_1\ \ \alpha_2\mathbf{z}_2 \cdots \alpha_n\mathbf{z}_n]$, where the α_is are positive real numbers.*

Proof : $\mathbf{z}_i \geq 0$ implies $\alpha_i\mathbf{z}_i \geq 0$. $A(\alpha_i\mathbf{z}_i) = \alpha_i A\mathbf{z}_i = 0$; i.e., P1 is satisfied. Further, if $\sum_{i=1}^{n} \gamma_i(\alpha_i\mathbf{z}_i) = 0$, then since the vectors $\{\mathbf{z}_i\}$ are linearly independent, $\gamma_i\alpha_i = 0$, which implies $\gamma_i = 0$. If $D' = \text{diag}(\alpha_1, \alpha_2, \ldots, \alpha_n)$, then $Z' = ZD'$. Clearly, $X' = XD'$ where X, X' are the corresponding input connectivity matrices. The principal submatrices of X' are given by the principal submatrices of X multiplied by a appropriate diagonal matrix. Hence, the sign of a principal submatrix, indeed of any term in the determinant expansion is preserved. This shows that P4/P4´is satisfied. If the diagonal matrix $D > 0$, it is invertible and hence XD has the same eigenvalues as DX. Thus, the eigenvalues of DXD' are the same as those of $XD'D = XD''$, where the diagonal matrix $D'' = D'D > 0$. By definition, XD' is also a D-stable matrix. Hence, P5 is satisfied. □

Interpretation: The theorem states that multiplying all the weights connected to a particular voltage V_i by some constant does not change the circuit behavior if the effects of incompletion are neglected.

Lemma 3.5.1 *If $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n\}$ is linearly independent and $\mathbf{x}_i = \mathbf{z}_i([1:n])$, the relation $\mathbf{1}_n = \sum_{i=1}^{n} \gamma_i\mathbf{x}_i$ holds for some γ_i. Then the set $\{\mathbf{z}'_1, \mathbf{z}'_2, \ldots, \mathbf{z}'_n\}$ is linearly independent, where*

$$\mathbf{z}'_i \ = \ \beta_i\mathbf{1}_m \ + \ \mathbf{z}_i \qquad i = 1, 2, \ldots, n, \tag{3.8}$$

if the β_is satisfy $1+\sum_i \beta_i\gamma_i \neq 0$. In particular, if $\gamma_i \geq 0$, then the β_is can be any nonnegative real number.

Proof : Let $\sum_{i=1}^{n} \alpha_i\mathbf{z}'_i = 0$. This implies that $\sum_{1}^{n} \mathbf{x}'_i = 0$, where $\mathbf{x}'_i = \mathbf{z}'_i([1:n])$. Then

$$\begin{aligned}\sum_{i=1}^{n} \alpha_i\mathbf{x}'_i = \mathbf{1}_n \sum_{i=1}^{n} \alpha_i\beta_i + \sum_{i=1}^{n} \alpha_i\mathbf{x}_i \\ = \sum_{i=1}^{n}(c\gamma_i + \alpha_i)\mathbf{x}_i \ \text{, where } c = \sum_{i=1}^{n} \alpha_i\beta_i.\end{aligned}$$

Since the vectors $\{\mathbf{z}_i\}$ are linearly independent, it follows that the vectors $\{\mathbf{x}_i\}$ are linearly independent, and hence $\alpha_i = -c\gamma_i$. Therefore, $c = -c\sum_{i=1}^{n} \gamma_i\beta_i$, and hence $c(1 +$

$\sum_{i=1}^{n} \gamma_i \beta_i) = 0$. By the conditions of the theorem, $1 + \gamma_i \beta_i \neq 0$, which implies that $c = 0$. Hence, $\alpha_i = -c\gamma_i = 0$, which means that $\{\mathbf{z}'_i\}$ is a linearly independent set. □

Interpretation: The theorem states that adding a constant weight to all the weights connected to a voltage V_i does not change the circuit behavior apart from the effects of incompletion, as before.

Theorem 3.5.2 (Completion Theorem) *If $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}$ with $\mathbf{z}_i \in \mathbb{N}^m$ are such that*

A1 $A\mathbf{z}_i = 0 \qquad i = 1, 2, \ldots, n-1,$

A2 The vectors $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}, \mathbf{1}_m\}$ are linearly independent,

then $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n$ satisfies R1, R2, and R3 with $w = \|\sum_{i=1}^{n-1} \mathbf{z}_i\|_\infty \triangleq \max \sum_{i=1}^{n-1} \mathbf{z}_i$, where $\mathbf{z}_n = w\mathbf{1}_m - \sum_{i=1}^{n-1} \mathbf{z}_i$.

Proof : Let $S = \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n\}$. $\mathbf{z}_n \in \mathbb{N}^m$ by the definition of w. Clearly, $A\mathbf{z}_n = wA\mathbf{1}_m - \sum_{i=1}^{n-1} A\mathbf{z}_i = 0$, because of A1 and (3.2). Hence, S satisfies R1. R2 is valid by the definition of $\mathbf{z}_n$. To check R3, let $\sum_{i=1}^{n} \alpha_i \mathbf{z}_i = 0$. Using the definition of $\mathbf{z}_n$, this is equivalent to $\sum_{i=1}^{n-1}(\alpha_i - \alpha_n)\mathbf{z}_i + \alpha_n w \mathbf{1}_m = 0$. By A2, it follows that $\alpha_n w = 0$ and $\alpha_i - \alpha_n = 0$ for $i \in [1 : n-1]$. We can conclude that $w \neq 0$; for otherwise, all the $\mathbf{z}_i$ are zero, which contradicts A2. It is clear that $\alpha_n = 0$, which implies $\alpha_i = 0$. Hence, R3 is also satisfied by S. □

Hence, the problem of satisfying R1, R2, and R3 reduces to the problem of finding $\mathbf{z}_1, \mathbf{z}_2, \ldots \mathbf{z}_{n-1}$ satisfying A1 and A2. On the other hand, if n linearly independent vectors are already obtained, then the task of choosing $n-1$ vectors satisfying the conditions of the Completion Theorem is simplified by the following Corollary:

Corollary 3.5.1 *Let the set $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_n\}$ with $\mathbf{z}_i \in \mathbb{N}^m$ satisfy R1 and R3. Let $\mathbf{x}_i = \mathbf{z}_i([1 : n])$. Since the set $\{\mathbf{x}_i\}_{i=1}^{n}$ forms a basis for $\mathbb{R}^n$, there exist unique numbers γ_i so that $\mathbf{1}_n = \sum_{i=1}^{n} \gamma_i \mathbf{x}_i$. If $\gamma_k \neq 0$, then $\{\mathbf{z}_1, \ldots, \mathbf{z}_{k-1}, \mathbf{z}_{k+1}, \ldots, \mathbf{z}_n\}$ satisfies A1 and A2. The vector $\gamma = [\gamma_i]$ can be obtained as $X^{-1}\mathbf{1}_n$, where X is the input connectivity matrix corresponding to $Z = [\mathbf{z}_1 \ \mathbf{z}_2 \ldots \mathbf{z}_n]$.*

Proof : Let $\gamma_k \neq 0$. A1 is valid by R1, so only A2 need be shown. Let $\sum_{i\neq k}^{n} \alpha_i \mathbf{z}_i + \alpha_k \mathbf{1}_m = 0$. This implies $\sum_{i\neq k}^{n}(\alpha_i + \alpha_k \gamma_i)\mathbf{z}_i + \alpha_k \gamma_k \mathbf{z}_k = 0$. By R3, $\alpha_k \gamma_k = 0$ and $\alpha_i + \alpha_k \gamma_i = 0$ for

$i \in [1:n],\ \ i \neq k$. Since $\gamma_k \neq 0$, it follows that $\alpha_k = 0$ and hence $\alpha_i = 0$, which shows that A2 is satisfied. $\square$

Interpretation: This theorem is to be used for cases when the circuit topology obtained satisfies all the conditions except R2. Such networks in which the number of inputs to each MITE is not the same for all MITEs are called incomplete networks. The completion procedure that applies directly to the circuit itself is as follows:

1. Take initial $i = 1$.
2. For each MITE, remove all connections to V_i.
3. Sum all the remaining weights connected to each MITE. Find the maximum of all such sums and call it w_i.
4. Repeat steps 2 and 3 for each integer i from 1 to n.
5. Choose an index, say k, so that w_k is the least of all the w_is. This index k should satisfy the hypothesis of the theorem, namely $\gamma_k \neq 0$.
6. Removing all the previous connections to V_k, reconnect V_k back to each MITE, the corresponding new weight to each MITE being: w_k-(sum of all other weights to the MITE).

The network is complete now, with the number of inputs to each MITE being w_k. The difference between this theorem and the existing completion theorem [22] is that in the latter the requirement is that $w\mathbf{1}_m = \|\sum_{j=1}^{n}\mathbf{z}_j\|_\infty$. Since the $\mathbf{z}_j$s are nonnegative vectors, the completion theorem given here gives a smaller, or at most the same, number of inputs to each MITE.

Using the above theorems, a very simple synthesis procedure can be given [53]. This does not result, in general, in minimal fan–in networks, which is dealt with in Section 3.9. However, it has the advantage that while the minimal fan–in network suffers from the unavailability of effective D-stability tests, the simple algorithm results in a network that is D-stable.

3.6 Simple Synthesis Procedure

The procedure described in this section is applicable to only those STLEs that are in POPL form; i.e.,

$$I_{n+p} = \prod_{q=1}^{n} I_q^{\Lambda_{pq}} \qquad p = 1, 2, \ldots, l, \tag{3.9}$$

where the *power matrix* Λ_{pq} is given by the $\Lambda = YX^{-1}$. In general, $\Lambda \in \mathcal{M}_{l,n}(\mathbb{Q})$ and is related to the corresponding translinear loop matrix A by the following

$$A = [\Lambda \quad -I_l] \tag{3.10}$$

Since $A\mathbf{1}_m = 0$, it follows that $\Lambda\mathbf{1}_n = \mathbf{1}_l$. Conversely, for synthesis, the given power matrix Λ must have rational elements and must satisfy $\Lambda\mathbf{1}_n = \mathbf{1}_l$.

Though the following synthesis procedure is applicable to any general power matrix, for illustrative purposes, the function below is synthesized:

$$\begin{aligned} I_4 &= I_1 I_2^{1/2} I_3^{-1/2} \\ I_5 &= I_1^2 I_2 I_3^{-2} \end{aligned} \tag{3.11}$$

Clearly,

$$\Lambda = \begin{bmatrix} 1 & 1/2 & -1/2 \\ 2 & 1 & -2 \end{bmatrix}$$

In general, the POPL function is given as the $n \times l$ power matrix Λ. First, $m = n + l$ MITEs are drawn, the first n being called input MITEs and the next l being called the output MITEs. The drain voltage of the i^{th} input MITE is called V_i. The connectivity matrices are then obtained through the following steps:

1. Diode connect each input MITE. In other words, each V_i is connected to the corresponding input MITE i.e., the i^{th} MITE, through a unit weight. Connect each V_j to the $(n+i)^{\text{th}}$ MITE through a weight Λ_{ij}. This is equivalent to taking

   $$\tilde{Z} = \begin{bmatrix} I_n \\ \Lambda \end{bmatrix} \qquad \tilde{\mathbf{z}}_i = \begin{bmatrix} \mathbf{e}_i \\ \Lambda\mathbf{e}_i \end{bmatrix}$$

   where $\mathbf{e}_i$ is the $n \times 1$ unit column vector with 0 everywhere except at the i^{th} row. The MITE network for the example at this stage is shown in Figure. 3.2(a). Note that a zero weight represents no connection.

2. If any of the weights connected to the voltage V_j are negative, add a constant weight to each of the weights connected to V_j so that the weights are nonnegative. Repeat this for all the voltages. In effect, a new set of column vectors given by

$$\mathbf{z}_i = \beta_i \mathbf{1}_m + \begin{bmatrix} \mathbf{e}_i \\ \Lambda \mathbf{e}_i \end{bmatrix} \qquad i \in [1:n]$$

is defined where $\beta_i = \max(0, -\min(\{\Lambda_{ki} | k = 1, 2, \ldots l\}))$. In the example, it is clear that 2 should be added to all the weights connected to V_3 to make the weights nonnegative, which gives the MITE network in Figure. 3.2(b).

3. If the weights connected to the voltage V_j are not integers, multiply all the weights connected to a voltage V_j so that the weights become so. In terms of column vectors, if $\mathbf{z}_i$ has non-integer values as its components, multiply it by a suitably chosen α_i. Typically, α_i is the least common multiple of all denominators of the i^{th} column of Λ i. e., $\Lambda \mathbf{e}_i$. The new vectors are

$$\mathbf{z}'_i = \alpha_i \mathbf{z}_i = \beta_i \alpha_i \mathbf{1}_m + \begin{bmatrix} \alpha_i \mathbf{e}_i \\ \alpha_i \Lambda \mathbf{e}_i \end{bmatrix} \qquad i \in [1:n]$$

In this example, $\alpha_1 = 1; \alpha_2 = 2; \alpha_3 = 2$ so that the resultant MITE network is as in Figure. 3.2(c).

4. If the network is incomplete, use the completion theorem to get a complete network. Hence, the final choice of vectors is given by

$$\mathbf{z}''_i = \begin{cases} \beta_i \alpha_i \mathbf{1}_m + \begin{bmatrix} \alpha_i \mathbf{e}_i \\ \alpha_i \Lambda \mathbf{e}_i \end{bmatrix} & i \neq k \\ (w - \sum_{j \neq k}^{n} \beta_j \alpha_j) \mathbf{1}_m - \sum_{j \neq k}^{n} \begin{bmatrix} \alpha_j \, \mathbf{e}_j \\ \alpha_j \, \Lambda \mathbf{e}_j \end{bmatrix} & i = k \end{cases} \tag{3.12}$$

In our example, applying the theorem, the smallest w is achieved for $k = 3$, for which $w = 4$. Our final MITE network in Figure. 3.2(d) has total number of weights to each MITE equal to 4.

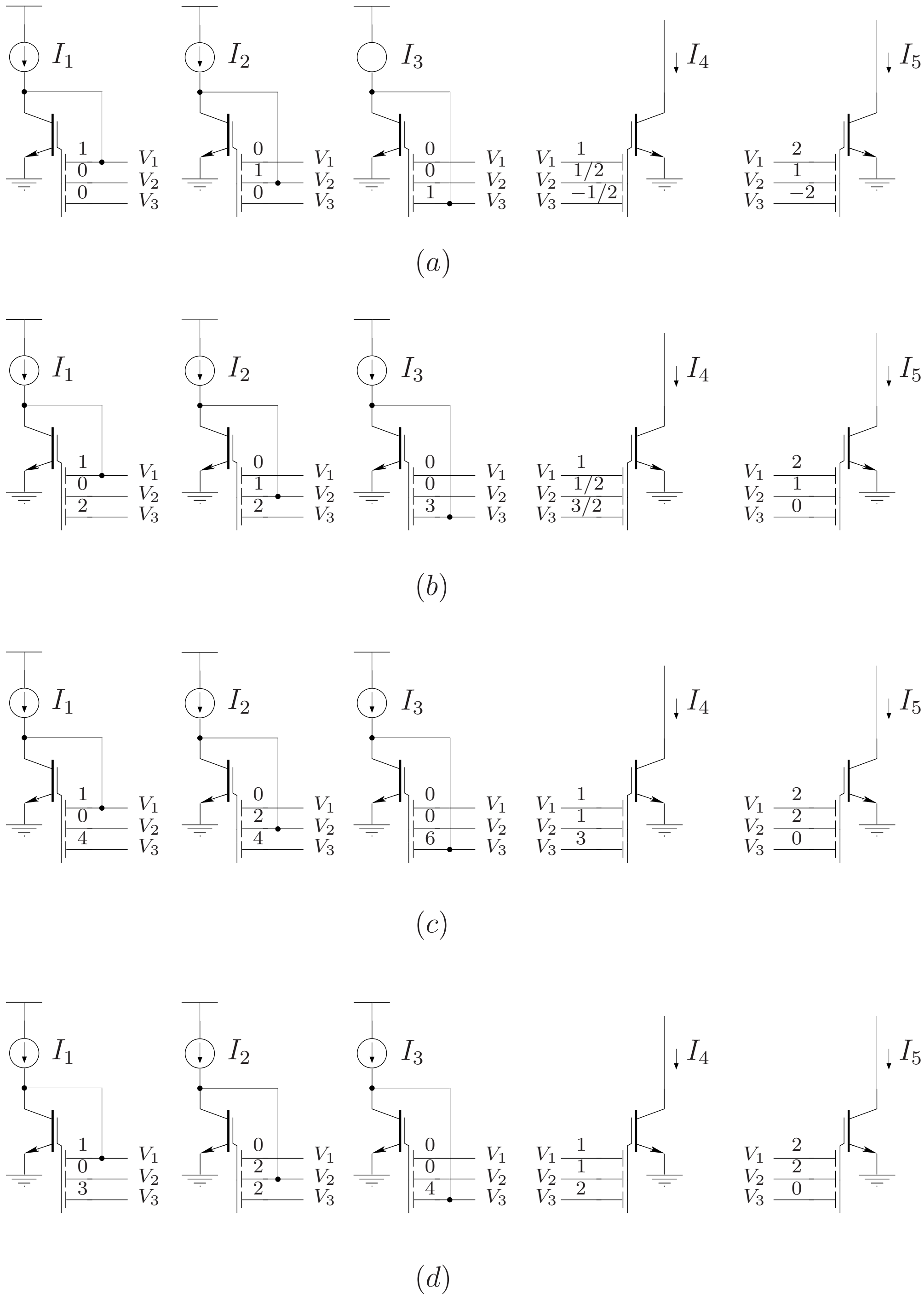


Figure 3.2. Synthesis of the MITE network implementing $I_4 = I_1 I_2^{1/2} I_3^{-1/2}$; $I_5 = I_1^2 I_2 I_3^{-2}$ described in steps $1-4$. The MITE network in (a) is obtained by assuming the input connectivity matrix to be the identity and the output connectivity matrix to be Λ. The weights in (a) are rendered nonnegative by adding a weight 2 to all weights connected to V_3, which results in the network in (b). The nonnegative weights in (b) are converted into nonnegative integers by multiplying all weights connected to V_2 and V_3 by 2, which results in (c). The final network in (d) is obtained by using Theorem 3.5.2 i.e., the completion theorem.

3.6.0.2 Justification

The vectors $\mathbf{e}_i$ are linearly independent, and hence the vectors $\tilde{\mathbf{z}}_i = \begin{bmatrix}\mathbf{e}_i \\ \Lambda\mathbf{e}_i\end{bmatrix}$ are linearly independent. Further, as $\mathbf{1}_n = \sum_{i=1}^n \mathbf{e}_i$, and $\Lambda\mathbf{1}_n = \mathbf{1}_l$, $\mathbf{1}_m = \sum_{i=1}^n \tilde{\mathbf{z}}_i$ and hence the conditions of Lemma 3.5.1 are satisfied with $\gamma_i = 1 > 0$. Hence, the choice of $\mathbf{z}_i$ in step 2 satisfies R3. Further, β_i in step 2 is chosen so that $\mathbf{z}_i$ are nonnegative. In step 3, the multiplication by α_i is justified by Theorem 3.5.1 and the choice of α_i makes sure that $\mathbf{z}_i \in \mathbb{N}^m$. Theorem 3.5.1 is now applicable because for any choice of k in step 4,

$$
\begin{aligned}
(\sum_{i=1}^{n} \beta_i + 1)\mathbf{1}_m &= \sum_{i=1}^{n} \frac{1}{\alpha_i}\,\mathbf{z}'_i \\
\therefore \quad \gamma_i &= \frac{1}{\alpha_i}\Big(\frac{1}{\sum_i \beta_i + 1}\Big) \neq 0, \quad i \in [1:n]
\end{aligned}
$$

Hence, k can be chosen to be any integer between 1 and n.

3.6.0.3 Stability

For proving the D-stability of X, the following theorem is needed:

Theorem 3.6.1 *Given that $\tau_i > 0$, $\eta_i, \xi_i \geq 0$, all the zeros of the polynomial $g(\lambda)$ have negative real parts where*

$$
g(\lambda) \;=\; \prod_{j=1}^{n}(\lambda\tau_j + 1) + \sum_{\substack{i=1\\ i\neq k}}^{n} \Big\{(\lambda\tau_i\eta_i + \xi_i) \prod_{\substack{j=1\\ j\neq k,i}}^{n} (\lambda\tau_j + 1)\Big\}
$$

Proof : The theorem is proved for the case when the τ_is are distinct and when $\eta_i \neq \xi_i$; the remaining cases can be easily shown to reduce to the this case. It is clear that for this case $\lambda = -1/\tau_i$ is not a root of $g(\lambda) = 0$ and hence the roots of $g(\lambda) = 0$ coincide with those of $g(\lambda)/\prod_{j\neq k}(\lambda\tau_j + 1) = 0$. If $\sigma = \Re e(\lambda)$, then

$$
\Re e\Big(\frac{g(\lambda)}{\prod_{j\neq k}(\lambda\tau_j + 1)}\Big) = \sigma\tau_k + 1 + \sum_{i\neq k} \frac{|\lambda\tau_i|^2\eta_i + \sigma\tau_i(\xi_i + \eta_i) + \xi_i}{|\lambda\tau_i + 1|^2}
$$

Therefore, if $\sigma \geq 0$, $\Re e(g(\lambda)/\prod_{j\neq k}(\lambda\tau_j + 1)) > 0$. Hence, if $\sigma \geq 0$, $g(\lambda) \neq 0$. □

Theorem 3.6.2 *The input connectivity matrix X of the MITE network obtained by the synthesis procedure described in steps 1-4 is D-stable for any choice of k in step 4.*

Proof : Let X be the input connectivity matrix with column vectors $\mathbf{x}''_i = \mathbf{z}''_i([1:n])$ as given in Equation 3.12 and T be the diagonal matrix with diagonal elements τ_i. By properties of determinants,

$$
\begin{aligned}
det(\lambda T + X) &= (\lambda\tau_k + W)\prod_{i\neq k}^{n}(\lambda\tau_i + \alpha_i) + \sum_{i\neq k}^{n}\left\{\alpha_i\beta_i\lambda(\tau_k - \tau_i)\prod_{j\neq k,i}^{n}(\lambda\tau_j + \alpha_j)\right\} \\
&= \gamma\Big[\prod_{j=1}^{n}(\lambda\tau_j' + 1) + \sum_{i\neq k}^{n}\left\{(\lambda\tau_i'\eta_i + \xi_i)\prod_{j\neq k,i}^{n}(\lambda\tau_j' + 1)\right\}\Big]
\end{aligned}
$$

where $\gamma = (W - \sum_{i\neq k}\alpha_i\beta_i)\prod_{j\neq k}^{n}\alpha_j$

$$
\tau_i' = \begin{cases} \tau_i/\alpha_i & i \neq k \\ \tau_k/(W - \sum_{i\neq k}\alpha_i\beta_i) & i = k \end{cases}
$$

$\eta_i = \beta_i\tau_k'/\tau_i'$; $\xi_i = \alpha_i\beta_i/(W - \sum_{j\neq k}^{n}\alpha_j\beta_j)$. Given that by construction, $W - \sum_{i\neq k}\alpha_i\beta_i \geq \alpha_j \geq 1$, the last equation satisfies the required conditions in Theorem 3.6.1. Hence, by Theorem 3.6.1, the characteristic polynomial of $-T^{-1}X$ has only roots with negative real parts. □

Since the objective is to minimize the fan–in w, it seems intuitively obvious that it suffices to "minimize" the $\mathbf{z}_i$ in some sense. This notion is made precise in the following.

3.7 Linear Diophantine Equations

Let us consider the *linear Diophantine equation*

Given $A \in \mathcal{M}_{l,m}(\mathbb{N})$

Problem Find $\mathcal{S} = \{\mathbf{z} \in \mathbb{N}^m |\ A\mathbf{z} = 0\}$

There exists a finite subset $\mathcal{H}$ of the solution set $\mathcal{S}$, called the *Hilbert basis* or the set of *minimal solutions* of the diophantine equation, such that every element of $\mathcal{S}$ can be written as a nonnegative integral combination of the elements of $\mathcal{H}$. The elements of $\mathcal{H}$ are minimal in the sense that if $\mathbf{u} \in \mathcal{H}$, then there is no other $\mathbf{v} \in \mathcal{S}$, $\mathbf{v} \neq 0$ such that $\mathbf{u} \gg \mathbf{v}$, where by definition, for some $m \times 1$ vectors $\mathbf{a} = [a_i]$ and $\mathbf{b} = [b_i]$, $\mathbf{a} \gg \mathbf{b}$ means that $a_i \geq b_i$ for all indices i with strict inequality for at least one index.

Various algorithms exist for finding the set of minimal solutions [54]. The algorithm used here is the so-called ABCD algorithm [55, 56]. A simplified description of the algorithm is given here.

In the case of a single equation $A\mathbf{x} = \sum_{i=1}^{m} a_i x_i = 0$, the search for minimal solutions can be done using the algorithm below, due to Fortenbacher: start with the standard basis for $\mathbb{N}^m$, and if at some stage $\mathbf{x} = [x_i] \in \mathbb{N}^m$ is not a solution,

if $\sum_{i=1}^{m} a_i x_i < 0$ and $a_j > 0$, then increment x_j by 1.

if $\sum_{i=1}^{m} a_i x_i > 0$ and $a_j < 0$, then increment x_j by 1.

This can be written as

C1 If $A\mathbf{x}.A\mathbf{e}_j < 0$, then increment x_j by 1.

If after incrementing, $\mathbf{x}$ is greater than (i.e., $\gg$) any previous solution, then it is removed. Of course, if $A\mathbf{x} = 0$, it is added to the minimal solution set. The ABCD algorithm extends the previous algorithm to systems of linear equations by applying the same restriction C1 for the case when A is a matrix, with $A\mathbf{x}.A\mathbf{e}_j$ interpreted as a scalar product of $A\mathbf{x}$ and $A\mathbf{e}_j$. It is shown that the process stops after a finite number of steps and that all the minimal solutions, and only the minimal solutions are found. The actual algorithm to be used here is a more efficient refinement of the above idea [55, 56].

3.8 Existence and Construction of Solution

The following theorem shows that minimal fan–in POPL MITE networks can be constructed using the vectors in the minimal solution set $\mathcal{H}$.

Theorem 3.8.1 (Construction Theorem)

1. *There exist vectors $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}$ with $\mathbf{z}_i \in \mathcal{H}$ satisfying A1 and A2 given in Theorem 3.5.2.*

2. *The minimum possible fan–in is also obtained as $w_{min} = \min\{\|\sum_{i=1}^{n-1} \mathbf{z}_i\|_\infty | \, \mathbf{z}_i \in \mathcal{H}\}$; i.e., the fan–in can be minimized by appropriately choosing elements of $\mathcal{H}$, which is a finite set compared to the solution set $\mathcal{S} = \{\mathbf{z} \in \mathbb{N}^m | \, A\mathbf{z} = 0\}$, which is infinite.*

Proof of 1: The proof proceeds in two steps. First, it is proved that $\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}$ can be chosen from $\mathbb{N}^m$. Next, it is shown that a solution exists in $\mathcal{H}$.

By Convention 3.1.1, $\text{rank}(A) = l$, hence $\text{nullity}(A) = m - l = n$. Since $A \in \mathcal{M}_{l,m}(\mathbb{Q})$, it can be considered as a linear transformation from $\mathbb{Q}^m$ onto $\mathbb{Q}^l$. Hence, $\{\mathbf{z} \in \mathbb{Q}^m | \, A\mathbf{z} = 0\}$, which is the corresponding kernel of this linear transformation, has dimension n. Since $A\mathbf{1}_m = 0$, a basis for $\{\mathbf{z} \in \mathbb{Q}^m | \, A\mathbf{z} = 0\}$ can be constructed by suitably appending $n-1$ more vectors $\mathbf{z}'_1, \mathbf{z}'_2, \ldots, \mathbf{z}'_{n-1}$. Hence, n linearly independent vectors $\mathbf{z}'_1, \mathbf{z}'_2, \ldots, \mathbf{z}'_{n-1}, \mathbf{1}_m$ from $\mathbb{Q}^m$ satisfying $A\mathbf{z} = 0$ are obtained. It is clear that by multiplying all the $\mathbf{z}'_i$s by the least common multipliers of their elements, it can be assumed that $\mathbf{z}'_i \in \mathbb{Z}^m$. Let $-c_i$ be the most negative integer amongst the components of $\mathbf{z}'_i$. The set $\{\mathbf{z}'_1 + c_1\mathbf{1}_m, \mathbf{z}'_2 + c_2\mathbf{1}_m, \ldots, \mathbf{z}'_{n-1} + c_{n-1}\mathbf{1}_m, \mathbf{1}_m\}$ is clearly a subset of $\mathbb{N}^m$ and can be easily shown to be linearly independent. Since $A(\mathbf{z}'_i + c_i\mathbf{1}_m) = A\mathbf{z}'_i + c_iA\mathbf{1}_m = 0$, it has been shown that $\{\mathbf{z}_i\}_{i=1}^{n-1} \subset \mathbb{N}^m$ satisfying A1 and A2 can be chosen.

By the definition of $\mathcal{H}$, each $\mathbf{z}_i$ constructed above can be written as a nonnegative linear combination of elements $\mathbf{v}_1, \mathbf{v}_2, \ldots \mathbf{v}_k$ of $\mathcal{H}$. Hence, $\mathbf{z}_i = \sum_{j=1}^{k} \alpha_{ij}\mathbf{v}_j$, where $\alpha_{ij} \in \mathbb{N}$. Let $\mathbf{x}_i = \mathbf{z}_i([1:n])$ and $\mathbf{u}_i = \mathbf{v}_i([1:n])$. By R3, $\det[\mathbf{x}_1\, \mathbf{x}_2\, \cdots \mathbf{x}_{n-1}\, \mathbf{1}_n] \neq 0$, because of the linear independence of $\mathbf{z}_1, \ldots, \mathbf{z}_{n-1}$. Since the determinant is a linear function of each of the column vectors, $\det[\mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_{n-1}\mathbf{1}_n]$ can be written as a linear combination of determinants of the form $\det[\mathbf{u}_{i_1}\, \mathbf{u}_{i_2}\, \cdots\, \mathbf{u}_{i_{n-1}}\, \mathbf{1}_n]$, where $i_1, i_2, \ldots, i_{n-1}$ are integers between 1 and k. All these determinants cannot be zero, else $\det[\mathbf{x}_1\mathbf{x}_2 \cdots \mathbf{x}_{n-1}\mathbf{1}_n] = 0$. Hence, there exist vectors $\mathbf{v}_{i_1}, \mathbf{v}_{i_2}, \ldots, \mathbf{v}_{i_{n-1}}$ in $\mathcal{H}$ such that they satisfy A1 and A2. This proves part 1.

Proof of 2: Let $\{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}\} \subset \mathbb{N}^m$ satisfying A1 and A2 have the minimum possible fan–in; i.e., $\|\sum_{i=1}^{n-1} \mathbf{z}_i\|_\infty = w_{\min}$; such an element exists because of the well-ordering principle. If $\{\mathbf{z}_i\}_{i=1}^{n-1}$ is not a subset of $\mathcal{H}$, then since A1 is satisfied, each $\mathbf{z}_i = \sum_{j=1}^{k} \alpha_{ij}\mathbf{v}_j$, where $\alpha_{ij} \in \mathbb{N}$. Proceeding as in the previous part, it can be shown that for some $i_1, i_2, \ldots, i_{n-1}$, the vectors $\mathbf{v}_{i_1}, \mathbf{v}_{i_2}, \ldots, \mathbf{v}_{i_{n-1}}$ satisfy A1 and A2. However, since $\mathbf{v}_{i_j}$ is part of the nonnegative linear expansion of $\mathbf{z}_j$, it must be true that $\alpha_{ji_j} > 0$. Hence, $\mathbf{z}_j \gg \mathbf{v}_{i_j}$, which implies that $\sum_{j=1}^{n-1} \mathbf{z}_j \gg \sum_{j=1}^{n-1} \mathbf{v}_{i_j}$. Since, each of the vectors involved are nonnegative, it is clear that $w_{\min} \geq \|\sum_{j=1}^{n-1} \mathbf{v}_{i_j}\|_\infty$. By the definition of $w_{\min}$, $\|\sum_{j=1}^{n-1} \mathbf{v}_{i_j}\|_\infty = w_{\min}$. $\square$

The above theorem provides a method to generate MITE networks with minimum fan–in.

3.9 Optimal Synthesis Algorithm

Given $A \in \mathcal{M}_{l,m}(\mathbb{Z})$. $A(\beta, \gamma)$ is nonsingular.

Initalize the fan–in value w by using the fan–in obtained from the algorithm in [53]. Let the set of minimal connectivity matrices $\mathcal{V} := \emptyset$, initially.

Step 1 Find $\mathcal{H}$, the *finite* set of minimal solutions of $A\mathbf{z} = 0$ using the ABCD algorithm [55, 56].

Step 2 Choose $S' := \{\mathbf{z}_1, \mathbf{z}_2, \ldots, \mathbf{z}_{n-1}\} \subset \mathcal{H}$.

Step 3 Find the fan–in $w' := \|\sum_{i=1}^{n-1} \mathbf{z}_i\|_\infty$. If $w' > w$, go to Step 2.

Step 4 Check if S' satisfies A2. If no, go to Step 2 else use Theorem 3.5.2 to find $S := \{\mathbf{z}_i\}_{i=1}^n$ satisfying R1, R2 and R3.

Step 5 Check if a permutation σ of $[1:n]$ exists such that the matrix $Z := [\mathbf{z}_{\sigma(1)} \mathbf{z}_{\sigma(2)} \cdots \mathbf{z}_{\sigma(n)}]$ satisfies P4´. If not, go to Step 2. If yes, let $\mathcal{B}$ be the set of such Z matrices satisfying P4´.

Step 6 If $w' = w$, then $\mathcal{V} := \mathcal{V} \bigcup \mathcal{B}$. If $w' < w$, then $\mathcal{V} := \mathcal{B}$.

Step 7 If all possibilities of S' in Step 2 are not exhausted, repeat the sequence from Step 2.

Step 8 Check if $X = Z([1:n],[1:n])$ satisfies the sufficiency and necessary conditions for D-stability [47, 39] for all $Z \in \mathcal{V}$. If X is shown to be not D-stable, $\mathcal{V} := \mathcal{V} \setminus \{Z\}$.

3.10 Example

Let a MITE implementation be required for the STLE $I_1 I_2^{-2} I_3^2 I_6^{-1} = 1$; $I_1 I_2^{-2} I_3 I_5 I_7^{-1} = 1$; $I_1 I_2^{-2} I_4^2 I_8^{-1} = 1$, which is required in the construction of a rms-to-dc converter [18]. Here

$$A = \begin{bmatrix} 1 & -2 & 2 & 0 & 0 & -1 & 0 & 0 \\ 1 & -2 & 1 & 0 & 1 & 0 & -1 & 0 \\ 1 & -2 & 0 & 2 & 0 & 0 & 0 & -1 \end{bmatrix}$$

When the synthesis algorithm is used, the corresponding minimal solutions set $\mathcal{H}$, written as a matrix, and the corresponding MITE network connectivity matrix Z are

$$\mathcal{H} = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 2 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 2 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 2 \end{bmatrix} Z = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 2 & 0 \end{bmatrix}$$

Clearly, $w_{\min} = 2$. It can be verified that this Z satisfies P1-P5. Synthesis by means of other methods [18, 53] gives a non-minimal fan–in of 3.

3.11 Appendix 3.A

3.11.1 MATLAB code for the simple synthesis procedure in Section 3.6

```
function [Z1,W] = MITE(A)

%Purpose: Use old synthesis procedure to generate a set of connectivity

%         matrices Z1 and # of input gates in each MITE given by W, given

%         the translinear loop power matrix A

%

% Input: A: integer matrix satisfying sum(A,2)=0; represents TL power

%           matrix

%

% Output: Z1: Z1(:,:,k) is a connectivity matrix generated by the old

%             method producing the translinear loop equations

%         W: W=sum(Z1(:,;,k),2)


% convert the TL power matrix to product of power law equations format

[l,m]=size(A); n=m-l;

L=-A(:,n+1:m)\A(:,1:n);

Lnum=abs(det(A(:,n+1:m)))*L;
```

```
Lden=abs(det(A(:,n+1:m)))*ones(l,n);
G=gcd(Lnum,Lden);
Lnum=Lnum./G; Lden=Lden./G;

Z1=zeros(m,n);% output put to zero if eqn is dimensionally incorrect
W=0;
% check if equation is dimensionally correct
if A*ones(m,1)==zeros(l,1)
    lcmLden=Lden(1,:);
    for j=2:l
        lcmLden=lcm(lcmLden,Lden(j,:)); %get the lcm of ith col of Lden
    end
    %using the simple synthesis procedure
    Z=[eye(n);L];
    b=max(zeros(1,n),-min(Z,[],1));
    b1=ones(m,1)*b;
    lcm1=ones(m,1)*lcmLden;
    Z=Z+b1;
    Z=Z.*lcm1;
    W1=ones(1,n);
    for i=1:n
        W1(i)=max(sum(Z,2)-Z(:,i));
    end
    W=min(W1);
    IX=find(W1==W);
    s=0;
    %consolidation (balancing)
    for k=1:length(IX)
        B=Z;
        B(:,IX(k))=W*ones(m,1)-sum(Z,2)+Z(:,IX(k));
```

```
        t=0;
        for j=1:s
            if Z1(:,:,s)==B, t=1; break; end
        end
        if t==0, s=s+1; Z1(:,:,s)=B; end
    end
    Z1=round(Z1);  W=round(W);
end
```

3.11.2 MATLAB code for finding the solution(s) Z given translinear loop matrix A

```
function Z=MITE_solve(A,Wmin,FILEN,opnomax)


%   Z=MITE_solve(A,Wmin,FILEN,opnomax)
%Purpose: takes the TL power matrix and gives the minimal connectivity matrices
%         with # of input gate greater than equal to Wmin. The input
%         connectivity matrix is required to be a P matrix.
%         Additionally, it also checks whether the output-input equations
%         are monotonic in the powers specified by C and outputs the
%         matrices satisfying monotonicity
% Inputs:
%      A:       integer matrix satisfying sum(A,2)=0;
%               input currents are specified first and then output currents
%               e.g the TL equations I_1^2 I_2 I_3^(-3)=1 and I_1 I_2 I_4^(-2)=1
%               where I_1 and I_2 are the inputs. Here, A=[2 1 -3 0; 1 1 0 -2]
%
%   Wmin:       lower bound on # of input gates to each MITE
%               DEFAULT=2
% sol_no_max:   approximate number of solution matrices desired
%               DEFAULT=inf
```

```
%      FILEN:   number of solutions obtained after which the matrices are
%               stored in a data file
%               DEFAULT=10000
% Outputs:  Z: three-dimensional array such that Z(:,:,k) is a connectivity matrix
%              of a MITE network implementing the TL equation represented by A.
%              Z(i,j,k) is simply the weight connecting the voltage V_j to
%              the i th MITE (in the k th solution)
%              In Z(:,:,k), the rows (corresponding to each MITE) are numbered so that
%              input currents are given first and then the outputs

% solving for minimal non-negative integer vectors of Ax=0
B=contejean(A);
n=size(A,2)-size(A,1);
% giving default values to Wmin and C
if nargin<2, Wmin=1; end
% using the old synthesis procedure to get Wmax= an initial estimate of W
[Z,Wmax]=MITE(A);
%generate the Z matrix chhilbas= choose_hilbert_basis
if nargin<5,opnomax=inf;end
if nargin<4, FILEN=10000;end
[Z,W]=chhilbas(B,n,Wmax,Wmin,FILEN,opnomax);
```

3.11.3 MATLAB code for finding the hilbert basis of A

```
function B=contejean(A,ymax)
% Purpose: finds the minimal non-negative integer vectors x satisfying Ax=0
% Input A: integer matrix
% Output B: matrix each of whose column is a minimal vector satisfying Ax=0

U=A'*A;
[p,q]=size(A);
```

```
P=zeros(q,q); F=zeros(q,q);
for i=1:q
    P(q-i+1,i)=1; F(q-i+2:q,i)=ones(i-1,1);
end
B=[];
while ~isempty(P)
    y=P(:,end);  w=F(:,end);
    P(:,end)=[]; F(:,end)=[];
    if all(A*y==0), B=[B y]; continue;  end
    s=0;
    for i=1:size(B,2)
        if vgeq(y,B(:,i)), s=1; break; end
    end
    if s==1, continue; end
    for j=q:-1:1
        if ((w(j)==0) & (y'*U(:,j)<0))
            P=[P y];
            P(j,end)=y(j)+1;
            F=[F w];
            if nargin==2
                if (P(j,end)==ymax(j)), F(j,end)=1;end
            end
            w(j)=1;
        end
    end
end
```

3.11.3.1 MATLAB code for vgeq:

```
function q=vgeq(x,y)
% Inputs: x and y are vectors of the same dimension
```

```
% q=1 if the elements of x-y are nonnegative with at least one positive
% element
if length(x)~=length(y)
    q=2;
    return % 2=error
end
if any(x<y), q=0; return; end
if any(x>y), q=1; return; end
q=0;
```

3.11.4 MATLAB code for forming solution matrices Z from hilbert basis

```
function [Z1,W]=chhilbas(B,n,Wmax,Wmin,sol_no_max,FILEN)
%  [Z1,W]=chhilbas(B,n,Wmax,Wmin,sol_no_max,FILEN)
% Purpose: To choose vectors from B to form a consolidated MITE network(s)
%          such that the input connectivity matrix is a P matrix
% Inputs:
%  B:      matrix whose columns are all minimal vectors of the linear
%          diophantine equation
%  n:      The number of input currents in the MITE network= # of columns of the
%          translinear loop matrix -# of rows of the translinear loop matrix
%  Wmax:   initial estimate on # of input gates to each MITE i.e there
%          should exist a solution with this no of input gates
%  Wmin:   lower bound on # of input gates to each MITE
%          The first two inputs are necessary; the default values of
%          the last two are Wmin=2 and Wmax=max(sum(B,2))
%  sol_no_max:   approximate number of solution matrices desired
%  FILEN:        number of solutions obtained after which the matrices are
%                stored in a data file
% Outputs: Z:           3-d array such that Z(:,:,k) is a solution matrix
% Convention: For translinear loop minimal vectors, the the input currents
```

```
%              are given first and then the outputs (in a column)

% Initialization
if nargin<6, FILEN=10000; end
if nargin<5, sol_no_max=inf;end
if nargin<4, Wmin=1; end
if nargin<3, Wmax=max(sum(B,2)); end
if Wmax<Wmin, Wmax=Wmin; end
[m,t]=size(B);
umax=(t-n+2):t;
u=1:(n-1);
W=Wmax;
v=1:n;
P=perms(v); q=factorial(n);
V=[]; V1=[];

% loop going through the (t choose n-1) vectors to check for
% invertibility/unique operating point property of the resulting consolidated
% MITE network
while (u(1)<=(t-n+2))
    W1=max(sum(B(:,u),2));
    X=B(1:n,u);
    %rank condition check
    if ((W1<=W) && (Wmin<=W1)) && (det([X ones(n,1)])~=0)
        if W1<W
            V=u;
            W=W1;
        else
            V=[V;u];
        end
```

```
    end
    if u(n-1)<t      %Case (1) u(n-1)<t
        u(n-1)=u(n-1)+1;
        continue
    else             % Case (2) u(n-1)=t
        s=max(find(umax-u));
        if isempty(s), break; end
        if isequal(s,1), u(s)+1, end
        u(s)=u(s)+1;
        u(s+1:n-1)=(u(s)+1):(u(s)+n-s-1);

    end
end
save('chhilbasint.mat','V');
size(V)
count=0;
ind=0;
while ~isempty(V)
    u=V(1,:);
    X=B(1:n,u);
    a=W*ones(n,1)-sum(X,2);
    X=[X a];
    u1=[u t+1];
    p=0;
    U=[];
    % check whether X is a permutation of a P matrix
    % when W=2, it is enough to check diag(X)>0 and det(X)~=0
    if W==2
        for j=1:q
            c=P(j,:);
```

```
            if all(diag(X(:,c))>0)
                p=p+1;
                U(p,:)=u1(c);
            end
        end
    else
        for j=1:q
            c=P(j,:);
            if any(diag(X(:,c))<=0) || (det(X(:,c))<=0), continue; end
            if Ptest(X(:,c))
                p=p+1;
                U(p,:)=u1(c);
            end
        end
    end
    V(1,:)=[];
    V1=[V1;U];
    count=count+size(U,1);
    if (isfinite(sol_no_max)) && ((FILEN*(ind)+count)==sol_no_max), break;end
    if count==FILEN
        ind=ind+1
        save(['chhilbasmat',num2str(ind),'.mat'],'V1');
        V1=[];
        count=0;
    end
end
V=V1;
for i=ind:-1:1
    load(['chhilbasmat',num2str(ind),'.mat'],'V1');
    V=[V1;V];
```

```
end
clear V1 U p P X c q u1 a u s W1 count ind
% Finding the connectivity matrices from the matrix of indices
B(:,t+1)=zeros(m,1);
k=1;
Z1=[];
for i=1:size(V,1)
    Z2=B(:,V(i,:));
    w=find(V(i,:)==t+1);
    if ~isempty(w)
        Z2(:,w)=W*ones(m,1)-sum(Z2,2);
    end
    tst=0;
    for j=1:k-1
        if isequal(Z2,Z1(:,:,j)), tst=1; break; end
    end
    if tst==0, Z1(:,:,k)=Z2; k=k+1; end
end
```

For testing whether a matrix is a P-matrix or not, we use the recently developed test by Tsatsomeros, the MATLAB code of which is given at
`http://www.sci.wsu.edu/math/faculty/tsat/files/matlab/ptest3.m`)

3.12 Conclusion

A new synthesis procedure for implementing systems of translinear–loop equations using MITEs is presented. This procedure results in minimal number of MITEs and the minimal obtainable fan–in for the minimum number of MITEs. The relationship between minimal fan–in of MITE networks and minimal solutions of linear Diophantine equations is shown. The resulting MITE networks have a unique operating point and their unconditional stability is tested with available methods.

CHAPTER 4

SYNTHESIS OF 2-MITE POPL NETWORKS

Any translinear circuit, at the fundamental level, requires the synthesis of translinear loops. Mathematically speaking, the synthesis of the following set of equations is required:

$$I'_i = \prod_{j=1}^{n} I_j^{\Lambda_{ij}}, \quad i = 1, 2, \ldots, l \tag{4.1}$$

where $\sum_{j=1}^{n} \Lambda_{ij} = 1$. A standard circuit called the product–of–power–law (POPL) network, shown in Figure 4.1(a), is used to implement these kinds of equations [19]. Two features of this network contributing to its size are the *number of MITEs* and the *number of input gates i.e., the fan–in, of a MITE*. The requirement of κ being the same for all MITEs translates to all the MITEs in a MITE network having the same fan–in [11]. Synthesis procedures that aim at reducing the number of MITEs are described in [53, 57]. This chapter, in contrast, concentrates on MITE networks with the minimum possible fan–in, namely 2. MITE circuits designed using the ideal expressions do not always have unique or stable operating points [51, 12]. These properties are shown to be automatically satisfied for 2-MITE POPL networks under some mild assumptions in Section 4.2. 2-MITE POPL networks are then analyzed using a graph-theoretic formulation and shown to belong to a particular class of digraphs in Section 4.3. It is shown that the uniqueness and stability of the operating point can be decided simply by counting the number of edges in directed

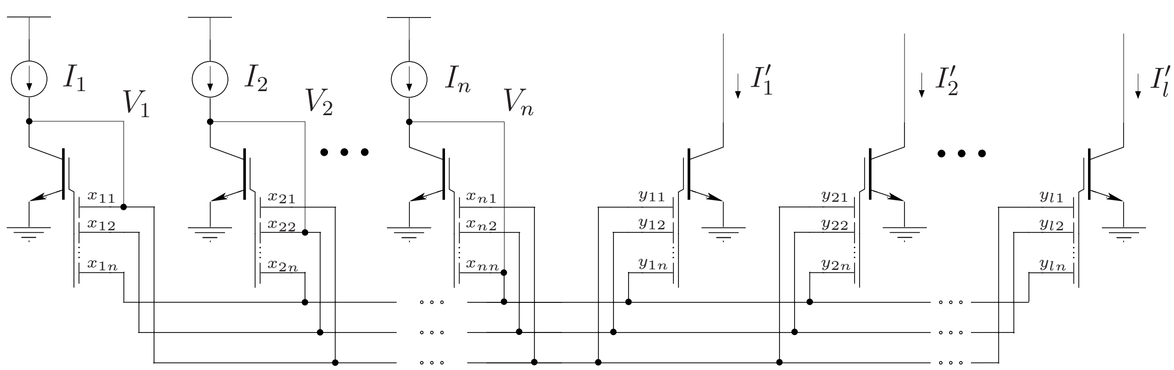


Figure 4.1. The general form of the MITE network implementing a POPL function. The output currents are a product of the input currents raised to different powers.

circuits in the digraph. The inverse of the input connectivity matrix X and the power-matrix is determined from the digraph itself. Necessary conditions for a power-matrix Λ to be implementable as a 2-MITE POPL network are then developed in Section 4.4. They are extended to sufficient conditions in the case of a POPL MITE network with a single output in Section 4.5. The synthesis of arbitrary POPL equations using 2-MITE networks with minimal number of MITEs used is discussed in Section 4.6. 2-MITE POPL networks in a reconfigurable framework are dealt with in Section 4.7. An ideal basic structure for use in the MITE FPAA is discussed here. A general Coates graph analysis that is expected to pave the way for the synthesis of multiple-output 2-MITE POPL networks is given in Section 4.8 followed by a detailed catalog of graphs corresponding to 2-MITEable POPL functions with two outputs in Section 4.9.

4.1 Mathematical Preliminaries

The terminology for directed graphs (digraphs) used here mostly follows [58]. A 1-factor of a digraph G is a spanning subgraph of G which is regular of degree 1 (i.e., both in-degree and out-degree is 1 for all vertices). A 1-factorial connection from i to j of a digraph G is a spanning subgraph G which contains a directed path P from i to j and a set of vertex-disjoint directed circuits that include all the vertices of G other than those in P. If x and y are vertices in a directed path P such that there is a directed subpath from x to y, then xPy denotes this subpath and $\bar{x}Py$ denotes the subpath from x to y excluding the initial vertex x. The weight $w(H)$ of a subgraph H of a weighted digraph G is the product of the weights of the edges in H.

4.2 Uniqueness and stability of Operating Point

In this section, we show that 2-MITE POPL networks have a unique and stable operating point under the assumption that its input connectivity matrix has a positive diagonal.

A POPL network is determined by the input-connectivity matrix $X = [x_{ij}]$ and the output connectivity matrix $Y = [y_{ij}]$, as shown in Figure 4.1(a). An input-output relationship of the form given by Equation (4.1) with $\Lambda = YX^{-1}$ results when X is nonsingular. In particular, a 2-MITE POPL network also satisfies

1. $X\mathbf{1}_n = \mathbf{2}\mathbf{1}_n$ and $Y\mathbf{1}_n = \mathbf{2}\mathbf{1}_l$, because the fan–in is two.

2. $x_{ij}, y_{ij} \in \{0, 1, 2\}$, since x_{ij} and y_{ij} are nonnegative integers.

The synthesis problem is the reverse process, that of finding suitable matrices X and Y given Λ. We will say that $\Lambda_{l\times n}$ or Equation (4.1) is *2-MITEable* if a 2-MITE POPL network satisfies Equation (4.1) *without using any copies* of the input currents i.e., the number of MITEs is $l+n$.

Ideally, the necessary and sufficient condition for the circuit in Figure 4.1(a) to have an unique operating point is "$\det(X) \neq 0$". The multiple feedback loops present in MITE circuits can, however, cause multiple operating points [51]. The following condition suffices to ensure that the operating point is unique:

P1 X is nonsingular and is a P_0-matrix, i.e., X has nonnegative principal minors.

This implies, in particular, that $x_{ii} \geq 0$.
A POPL MITE network described by the input-connectivity matrix X is stable in the sense of [12] if:

P2 X is D-stable, i.e., DX must be positive-stable for all diagonal matrices D with positive diagonal.

X satisfies **P**1 if it satisfies **P**2 [39]; however, there is no known finitely testable characterization for D-stability for matrices of order greater than three.

We will show that for 2-MITE networks, the following assumption suffices to satisfy both **P**1 and **P**2.

Assumption 1 *The input connectivity matrix X of a POPL network has a positive diagonal and is nonsingular.*

From Theorem 2.5.3, the above assumption along with the requirement of a unique operating point that is "robust" with respect to floating-gate capacitor mismatch leads to the following strengthened version of **P**1:

P1' X is a P-matrix.

For a 2-MITE POPL network, $x_{ii} (> 0)$ is either 1 or 2. Since the rows of X sum to 2, we can write $X = I_n + \widehat{X}$, where $\widehat{X}$ has exactly one nonzero entry, namely 1, in each row. Hence, for every row k, we can define a $\alpha(k)$ such that $[\widehat{X}]_{k\alpha(k)}$ is one. Clearly, X is *row diagonally dominant*, though not necessarily *strictly row diagonally dominant*. The following theorem then implies that X is D-stable.

Theorem 4.2.1 *If $A = [a_{ij}] \in \mathcal{M}_n$ is nonsingular, row-diagonally dominant, and has a positive diagonal, then A is D-stable. In particular, A is a P_0 matrix.*

Proof : Consider $DA = [d_i a_{ij}]$ where the diagonal matrix D has $d_{ii} > 0$. Gerv̌sgorin's theorem [39] tells us that the eigenvalues of DA lie in the union of n discs

$$G(DA) = \bigcup_{i=1}^{n} \{\lambda \in \mathbb{C} \,:\, |\lambda - d_i a_{ii}| \leq \sum_{j \neq i}^{n} |d_i a_{ij}|\} \tag{4.2}$$

The conditions $d_i a_{ii} > 0$ and $d_i a_{ii} \geq \sum_{j \neq i}^{n} |d_i a_{ij}|$ imply that each of the discs lies in the open right half s-plane with the possible exception of including 0. However, the case $\lambda(DA) = 0$ would imply that $\det(A) = 0$, which has been excluded by hypothesis. $\square$

The non-strict row-diagonal dominance property is not preserved under arbitrarily small perturbations of the elements of X and hence the above proof cannot be used to show that the D-stability is true in the presence of errors in the elements of X. It is shown in Appendix 4 that X is diagonally stable, which implies D-stability. Diagonal stability is maintained under small perturbations of the elements of X [49], and hence the D-stability is also preserved.

4.3 2-MITE network graphs

In this section, the structure of the Coates graphs of the input connectivity matrices of 2-MITE POPL networks is analyzed.

Restricting both the number of MITEs and the fan–in of a MITE also restricts the possible power matrices that are obtainable from a POPL network. If the fan–in is fixed at 2, it is necessary to find out which powers are obtainable before increasing the number of MITEs suitably. To this end, we take a graph theoretic approach to determine $\Lambda = YX^{-1}$

for a 2-MITE network. To find X^{-1}, we use the method of Coates graphs [58]. Every $A = [a_{ij}] \in \mathcal{M}_n$ corresponds to a weighted digraph $G_c(A)$ with vertices $\{1, 2, \ldots, n\}$ such that there is a directed edge (j, i) *from* j *to* i with weight a_{ij} if $a_{ij} \neq 0$. The graphical representation of the input-section of a 2-MITE network will then be the Coates graph $G_c(X)$ of its input-connectivity matrix X. As mentioned before, X can be written as $X = I_n + \widehat{X}$, where $\widehat{X}$ will be called the *reduced input-connectivity matrix* of the network. It should be noted that an equivalent simpler description of the input-section of a 2-MITE network is through the Coates graph $G_c(\widehat{X})$ of its reduced input-connectivity matrix, which is formed essentially by removing all the self-loops of unit weight in $G_c(X)$ and by converting a self-loop of weight 2 into a self-loop of unit weight. This is now illustrated through an example.

Example: Consider the MITE circuit in Figure 4.2(a). The input and output connectivity matrices X and Y are given by

$$X = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{bmatrix} Y = \begin{bmatrix} 0 & 0 & 0 & 0 & 2 \end{bmatrix} \tag{4.3}$$

The Coates graphs $G_c(X)$ and $G_c(\widehat{X})$ are shown in Figure 4.2(b) and (c), respectively. We will show in the later sections that the power matrix Λ can be found by inspection and is shown along in curly braces near each node.

Theorem 4.3.1 *If X is the input-connectivity matrix of a 2-MITE POPL network, then,*

1. *Each component G of $G_c(\widehat{X})$ has a unique directed circuit; self-loops being directed circuits of length* 1.

2. *If the directed circuit in the digraph G is contracted to a single vertex v, then the resulting digraph $\widetilde{G}$ is a rooted tree with v as the root i.e.,*

    - *The undirected graph underlying the digraph $\widetilde{G}$ is a tree.*

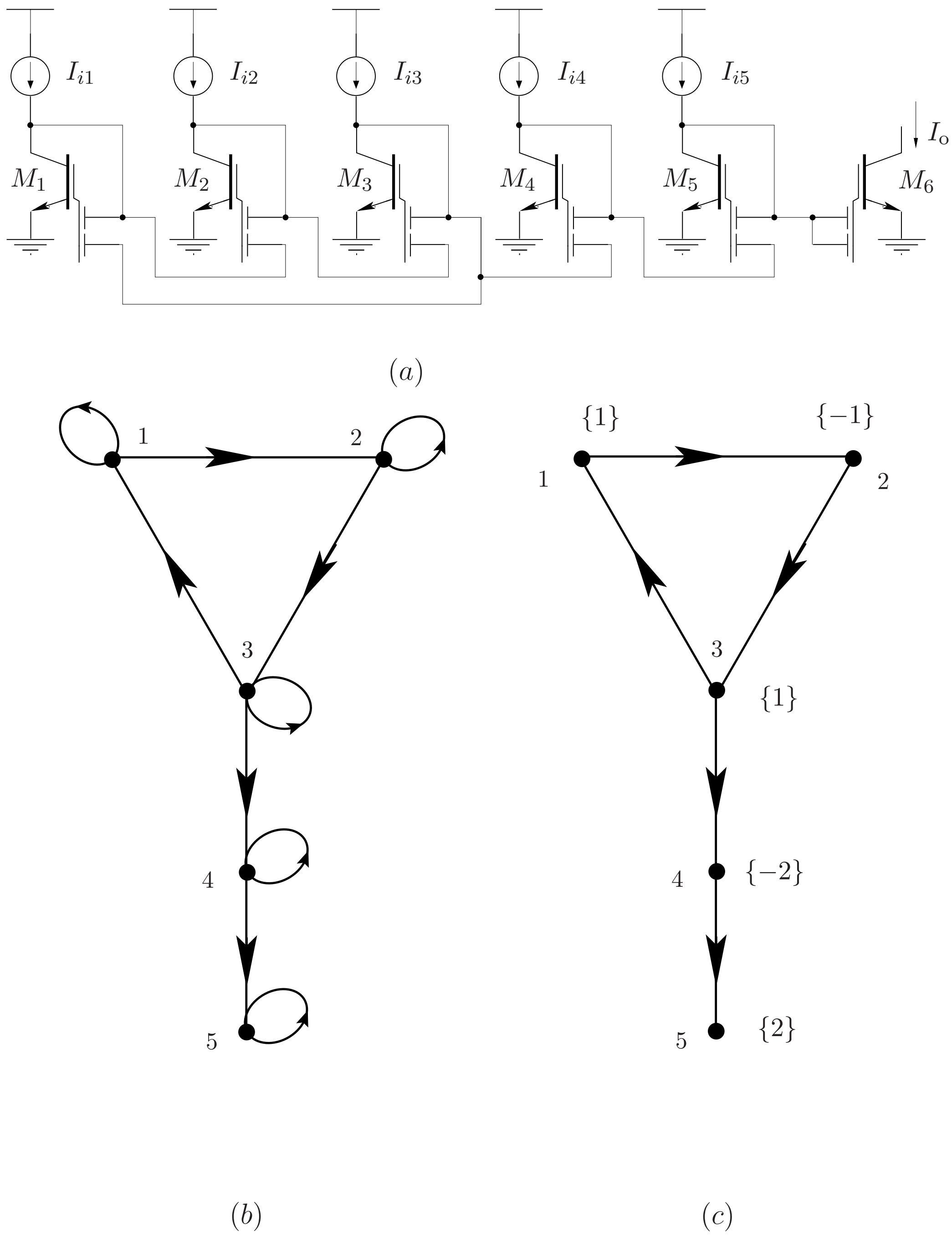


Figure 4.2. (a) A single-output 2-MITE network. (b) The Coates graph $G_{\rm c}(X)$ of the input–connectivity matrix X of the network (c) The Coates graph $G_{\rm c}(\widehat{X})$ of the reduced input-connectivity matrix $\widehat{X}$.

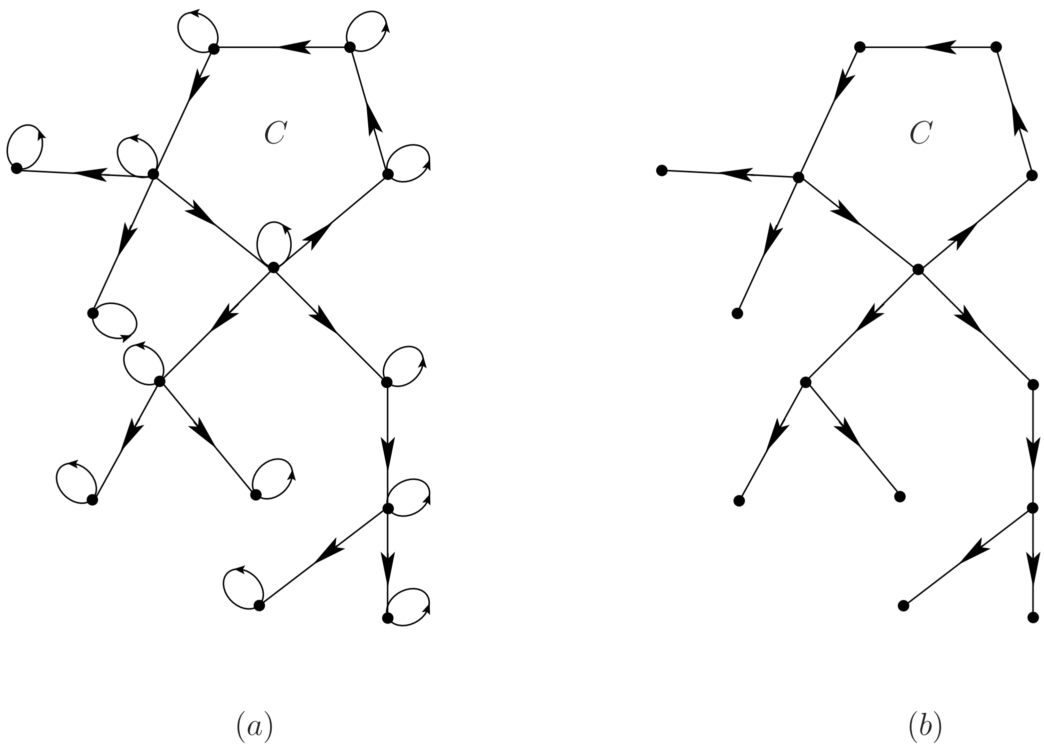


Figure 4.3. (a) A component of the Coates graph $G_c(X)$ of the input-connectivity matrix X of a 2-MITE POPL network with directed circuit C. (b) The Coates graph $G_c(\widehat{X})$ of the reduced input-connectivity matrix of the same network

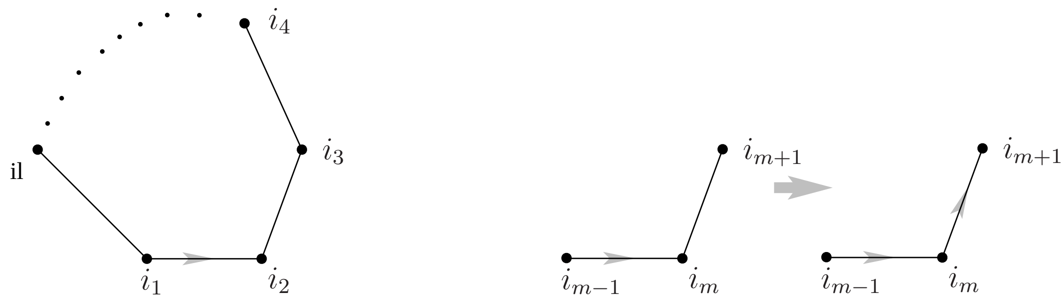
- *For every vertex $w \neq v$, there is a directed path from v to w.*

Proof of 1*:* Every vertex i in $G_{\mathrm{c}}(\widehat{X})$ has in-degree 1, and if $(j,i) \in E$, then $j = \alpha(i)$. Hence, we can define the *parent* $\alpha(i)$ and a sequence of *ancestors* $\{\alpha^k(i)\}$ for every vertex i. If vertex $j \neq i$ is an ancestor of vertex i, then we say $j \prec i$. We define $j \preceq i$ to mean that either $j \prec i$ or $j = i$. For any vertex i, consider the sequence $\{i, \alpha(i), \alpha(\alpha(i)), \dots\}$. Since there are only n vertices, the sequence cannot have distinct elements, and hence there exists a vertex j and an integer $p > 0$ such that $\alpha^p(j) = j$. This corresponds to a directed circuit of length *at most* p in $G_{\mathrm{c}}(\widehat{X})$ and implies that the sequence of ancestors of any vertex i eventually leads to a directed circuit C. It is easy to see that for a vertex i, there is only *one* directed circuit in $\{\alpha^k(i)\}$. We will say that i is *descended* from C.

The relation of being descendants of the same directed circuit is clearly an equivalence relation. We will show that the equivalence classes are the vertex sets of components of $G_{\mathrm{c}}(\widehat{X})$. If not, there is a undirected path P beginning from an equivalence class and ending in a different equivalence class. It is clear that there is an edge $(j,i) \in P$ where the vertices j and i belong to different equivalence classes. However, this implies that $j \prec i$ and hence j must be descended from the same directed circuit as i, which contradicts the definition of the classes. Each equivalence class being obviously connected, it follows that each is a component of $G_{\mathrm{c}}(\widehat{X})$.

Proof of 2*:* If the directed circuit in a component G is contracted into a single vertex v to form $\widetilde{G}$, it follows that the sequence of ancestors of any vertex i in G that was not in the directed circuit now ends at v. Hence, there is a directed path from v to each vertex in $\widetilde{G}$.

We will now show that $\widetilde{G}$ has a tree as its underlying graph. This is accomplished by showing that if there is a undirected circuit in G, then it must be a directed circuit. Since each component G is associated with an unique directed circuit which gets contracted in $\widetilde{G}$, this proves that there is no undirected circuit in $\widetilde{G}$. Let C' be a circuit in the graph underlying G as shown in Figure 4.4(a). Let (i_1, i_2) be the directed edge corresponding to an arbitrarily chosen edge in C'. Let $i_3, i_4, \dots, i_k$ be the remaining vertices in C' so that for every $s \in \{2, \dots, k-1\}$, exactly one of (i_s, i_{s+1}) and (i_{s+1}, i_s) is an edge in G. Also, exactly



(a) (b)

Figure 4.4. (a) The circuit C' used in the proof of 2 in Theorem 4.3.1. It is a hypothetical undirected circuit assumed to exist in the graph underlying G. The directed edge (i_1, i_2) is assumed to exist in G. (b) The proof shows that if we assume (i_{m-1}, i_m) to be a directed edge in G, then (i_m, i_{m+1}) has to be the directed edge connecting i_m and i_{m+1} in G.

one of (i_k, i_1) or (i_1, i_k) is an edge in G. Let (i_{m-1}, i_m) be in G. (i_{m+1}, i_m) cannot be an edge in G, since that would mean that the in-degree of i_m is not 1. Thus, we find that if (i_{m-1}, i_m) is an edge in G, then so is (i_m, i_{m+1}). We have chosen (i_1, i_2) to be in G. Hence, by induction, (i_m, i_{m+1}) is in G for $m \in \{1, 2, \ldots, l-1\}$. This argument can be extended to show that (i_k, i_1) must also be in G. Thus, we have proved that if C' is an undirected circuit in G, then the edges in C' must correspond to a directed circuit in G. Therefore, $\widetilde{G}$ has a tree for its underlying graph. From the definition of the rooted tree, this completes the proof. $\square$

This characterization of 2-MITE POPL networks enables us to find simple expressions for X^{-1}, as given below.

4.4 Necessary conditions

Using Coates graph analysis [58], we now derive expressions for X^{-1} and $\Lambda = YX^{-1}$. The determinant of $X \in \mathcal{M}_n$ is given by

$$\det(X) = \sum_H (-1)^{n-L_H} w(H) \tag{4.4}$$

where H is a 1-factor of $G_c(X)$, and L_H is the number of directed circuits in H. The cofactor Δ_{ij} of x_{ij} is given by

$$\begin{aligned}
\Delta_{ii} &= \sum_{H} (-1)^{n-1-L_H} w(H) \\
\Delta_{ij} &= \sum_{H_{ij}} (-1)^{n-1-L'_H} w(H_{ij}), \quad i \neq j
\end{aligned} \tag{4.5}$$

where H is a 1-factor in the graph obtained by removing i from $G_c(X)$, H_{ij} is a 1-factorial connection in $G_c(X)$ from vertex i to vertex j, and L_H and L'_H are the numbers of directed circuits in H and H_{ij}, respectively.

If $G_c(X)$ is not connected, then by reordering the rows and columns of X, we can write X as a direct sum of matrices X_i that are connected; each such matrix represents a component of $G_c(X)$. Since X^{-1} is the direct sum of the individual inverses, for finding X^{-1}, it suffices to assume that $G_c(X)$ is connected.

Some definitions are in order:

Definition 4.4.1 *When n is a nonnegative integer, we define $(-)^n$ to be $(-1)^n$. $(-)^\infty$ is defined to be* 0.

Definition 4.4.2 *The distance $d(i,j)$ is defined as the length of the shortest directed path* from *vertex j* to *vertex i, if a directed path exists. If no directed path exists from j to i, then $d(i,j)$ is defined to be ∞. $d(i,i)$ is defined to be* 0.

Theorem 4.4.1 *Let $G_c(\widehat{X})$ be the Coates graph of $\widehat{X}$, where $X = I_n + \widehat{X}$. If $G_c(X)$ is connected, and C is the unique directed circuit in $G_c(\widehat{X})$ with k edges in it, then*

$$\det(X) = 1 + (-1)^{k+1} \tag{4.6}$$

Clearly, X is nonsingular if and only if k is odd in which case $\det(X) = 2$. *X^{-1} is then given by*

$$[X^{-1}]_{ij} = \frac{[\mathrm{adj}(X)]_{ij}}{\det(X)} = \begin{cases} (-)^{d(i,j)} & \text{if } j \text{ is not in } C \\ \frac{1}{2}(-)^{d(i,j)} & \text{if } j \text{ is in } C, \end{cases} \tag{4.7}$$

where the distance $d(i,j)$ is defined with respect to either $G_c(\widehat{X})$ or $G_c(X)$.

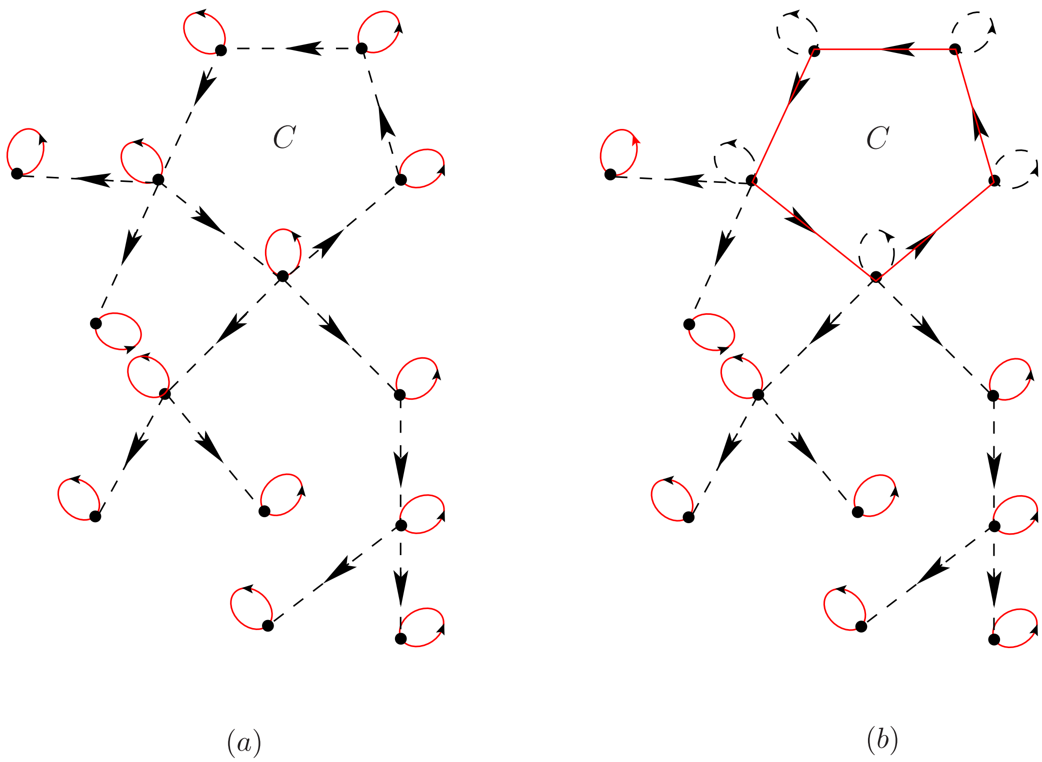


Figure 4.5. Calculation of $\det(X)$ using Equation (4.4). The edges in the 1-factors in $G_c(X)$ are shown with continuous edges while the edges not belonging to the 1-factor are shown by dotted edges. (a) The 1-factor formed by all self-loops in $G_c(X)$. (b) The 1-factor formed by C and the self-loops attached to vertices not in C.

Proof: The theorem will be proved for the case $k > 1$. The proof for the case when $k = 1$ i.e., when the directed circuit C is a self-loop, is almost identical and is left to the reader.

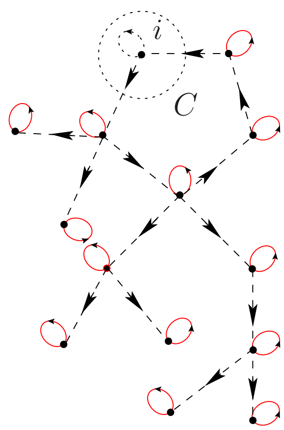
Calculation of $\det(X)$:

From the definition of a 1-factor, we need to find a set of vertex-disjoint circuits that include all the vertices in $G_c(X)$. As the only directed circuits in $G_c(X)$ are the self-loops attached to each vertex and the directed circuit C, it follows that there can be only two possible 1-factors in $G_c(X)$:

1. The set of all self-loops attached to each vertex in $G_c(X)$. This is shown in Figure 4.5(a).

2. The union of C and the set of all self-loops attached to each vertex in $G_c(X) \setminus C$, as shown in Figure 4.5.

In the first case, $w(H) = 1$ and $L_H = n$. In the second case, we still have $w(H) = 1$ but



(a) (b) (c)

Figure 4.6. Calculation of Δ_{ii} using Equation (4.5). The edges in the 1-factors in $G_c(X) \setminus i$ are shown with continuous edges while the edges not belonging to the 1-factor are shown by dotted edges. (a) When $i \in C$, all self-loops in $G_c(X) \setminus i$ form the only 1-factor. (b) When $i \notin C$, all self-loops in $G_c(X) \setminus i$ form one of the two 1-factors. (c) When $i \notin C$, the second 1-factor is formed by C and the self-loops in $G_c(X) \setminus i$ attached to vertices not in C.

$L_H = 1 + n - k$, since the self-loops of only $n - k$ vertices are taken into account. Hence, by using Equation (4.4), we get $\det(X) = (-1)^{n-n} \times 1 + (-1)^{n-(n-k+1)} \times 1 = 1 + (-1)^{k+1}$, proving the first part of the theorem.

Calculation of Δ_{ii}:

Case 1: $i \in C$

In this case, if we remove i from $G_c(X)$, the only 1-factor is the set of self-loops attached to all the remaining vertices, so that $w(H) = 1$ and $L_H = n - 1$. Hence, using Equation (4.5), we find that $\Delta_{ii} = (-1)^{n-1-(n-1)} \times 1 = 1$. This is shown in Figure 4.6(a).

Case 2: $i \notin C$

Here, if we remove i from $G_c(X)$, the two 1-factors are:

1. The set of all self-loops attached to each vertex in $G_c(X) \setminus i$. Here $w(H) = 1$ and $L_H = n - 1$. This is shown in Figure 4.6(b).

2. The union of C and the set of all self-loops attached to the vertices in $G_c(X) \setminus \{i, C\}$. Here $w(H) = 1$ and $L_H = n - 1 - k + 1 = n - k$. This is shown in Figure 4.6(c).

Hence, $\Delta_{ii} = (-1)^{n-1-(n-1)} + (-1)^{n-1-(n-k)} = 1 + (-1)^{k+1}$. Since X^{-1} exists only if k is odd, it follows that in that case, $\Delta_{ii} = 2$.

Calculation of Δ_{ji}:

Case 1: $j \in C$

See Figure 4.7(a). Here, it is clear that there is always a directed path from j to i, in accordance with Theorem 4.3.1. Hence there is only one 1-factorial connection, containing all the self-loops in the vertices that are not in the directed path. Since $d(i,j)+1$ is the number of vertices in the directed path from j to i, it follows that $L'_H = n - d(i,j) - 1$. Hence, from Equation (4.5), $\Delta_{ji} = (-1)^{n-1-(n-1-d(i,j))} \times 1 = (-1)^{d(i,j)}$.

Case 2: *$j \notin C$ and there is no directed path from j to i*

See Figure 4.7(b). In this case, $d(i,j) = \infty$. There is no 1-factorial connection from j to i and hence $\Delta_{ji} = 0 = 2(-)^{d(i,j)}$.

Case 3: *$j \notin C$ and there is a directed path from j to i*

See Figure 4.7(c) and (d). In this case, by Theorem 4.3.1, there is a unique directed path P_{ji}. Here, there are two 1-factorial connections:

1. The set of all self-loops attached to each vertex in $G_c(X) \setminus P_{ji}$. Here $w(H_{ij}) = 1$ and $L'_H = n - 1 - d(i,j)$.

2. The union of C and the set of all self-loops attached to the vertices in $G_c(X) \setminus \{P_{ji}, C\}$. Here $w(H_{ij}) = 1$ and $L'_H = n - (d(i,j)+1) - k + 1 = n - k - d(i,j)$.

Hence, $\Delta_{ji} = (-1)^{n-1-(n-1-d(i,j))} + (-1)^{n-1-(n-k-d(i,j))} = (-1)^{d(i,j)}(1 + (-1)^{k+1})$. When k is odd, we have $\Delta_{ji} = 2(-1)^{d(i,j)}$.

In general, it is clear that $\Delta_{ji} = 2(-)^{d(i,j)}$ if $j \notin C$ and $\Delta_{ji} = (-)^{d(i,j)}$ otherwise. Since $\det(X) = 2$ for a nonsingular X and $(X^{-1})_{ij} = \Delta_{ji}/\det(X)$, the conclusion stated by the theorem follows. $\square$

To find $\Lambda = YX^{-1}$, since $Y\mathbf{1}_n = 2\mathbf{1}_l$, it follows that every row in Y contains either a 1 in two different columns or a 2 in a single column; the other elements in the row being 0. Hence, we can write $y_{ij} = \delta_{j\beta(i)} + \delta_{j\gamma(i)}$, where $\beta(i)$ and $\gamma(i)$ denote the columns with

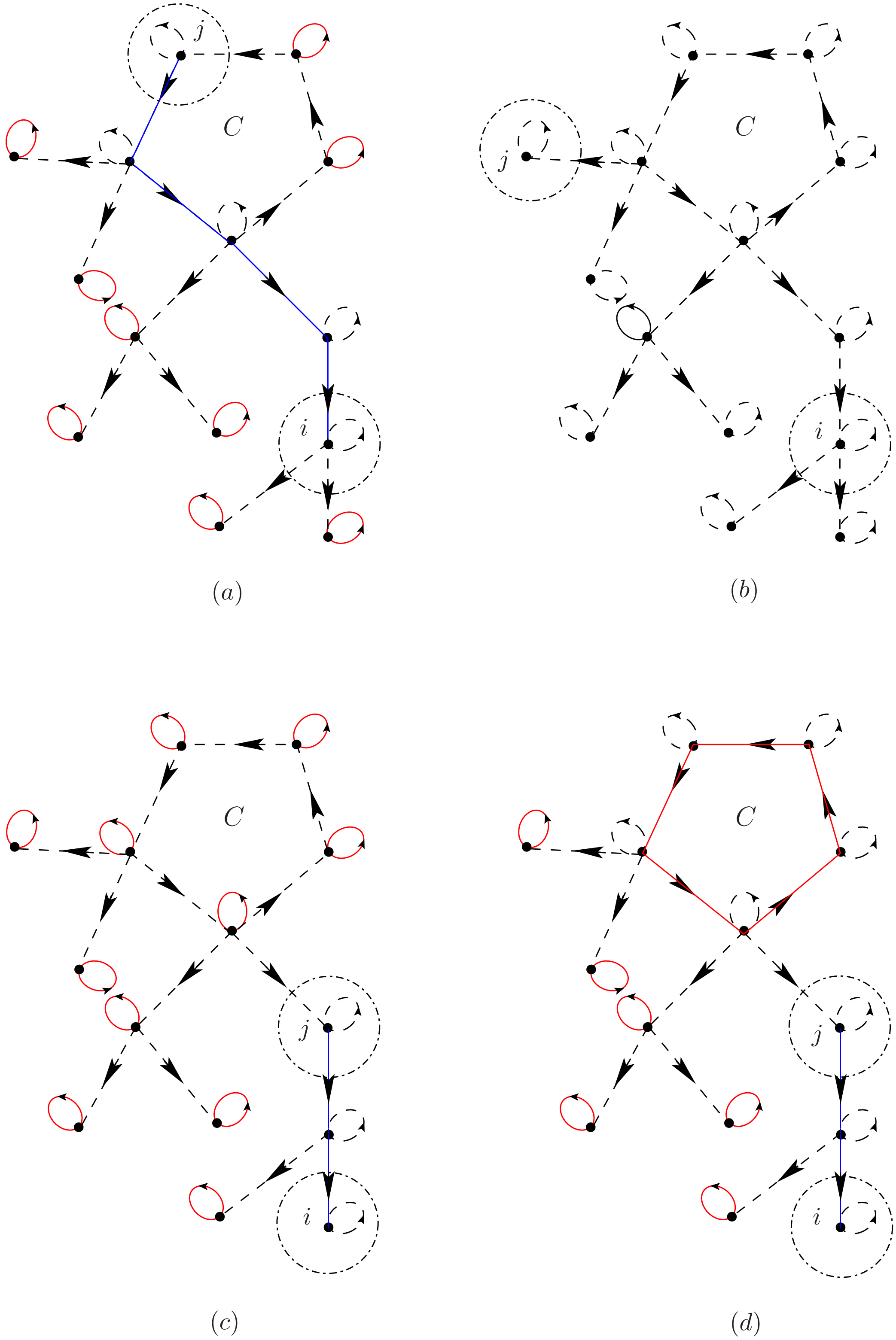


Figure 4.7. Calculation of Δ_{ji} using Equation (4.5). The edges in the 1-factorials involved are shown with continuous edges while the edges not belonging to the 1-factorials are shown by dotted edges. (a) When $j \in C$, P_{ji} and all self-loops in $G_c(X) \setminus P_{ji}$ form the 1-factorial. (b) When $j \notin C$ and there is no path from j to i, then no 1-factorial exists. (c) When $j \notin C$ and there is a P_{ji}, a 1-factorial is formed by P_{ji}, and the self-loops in $G_c(X) \setminus P_{ji}$. (d) When $j \notin C$ and there is a P_{ji}, another 1-factorial is formed by P_{ji}, C and the self-loops in $G_c(X) \setminus C \cup P_{ji}$.

nonzero entries in the i^{th} row of Y. From this, we have

$$\begin{aligned}
\Lambda_{ij} &= \sum_{k=1}^{n} y_{ik}[X^{-1}]_{kj} \\
&= \sum_{k=1}^{n} [\delta_{k\beta(i)}[X^{-1}]_{kj} + \delta_{k\gamma(i)}[X^{-1}]_{kj}] \\
&= [X^{-1}]_{\beta(i)j} + [X^{-1}]_{\gamma(i)j}
\end{aligned} \tag{4.8}$$

Thus, we have

$$\boxed{\Lambda_{ij} = \begin{cases} (-)^{d(\beta(i),j)} + (-)^{d(\gamma(i),j)} & \text{if } j \text{ is not in } C \\ \frac{1}{2}((-)^{d(\beta(i),j)} + (-)^{d(\gamma(i),j)}) & \text{if } j \text{ is in } C. \end{cases} \tag{4.9}}$$

The following general observations can be made from Equation (4.9): If Λ is 2-MITEable, then

- The only possible values for $(-)^{d(i,j)}$ are $-1, +1$ and 0. Hence it follows that

$$\begin{aligned}
&\Lambda_{ij} \in \{-2, -1, 0, 1, 2\}, \text{ if } j \text{ is not in } C \\
&\Lambda_{ij} \in \{-1, -\frac{1}{2}, 0, \frac{1}{2}, 1\}, \text{ if } j \text{ is in } C
\end{aligned} \tag{4.10}$$

- The same column in Λ cannot have both a ± 2 and a $\pm 1/2$. This is clear from the above. Thus, $\Lambda = \left(\begin{smallmatrix} 1 & -1 & .5 & .5 \\ -2 & 1 & 2 & 0 \end{smallmatrix}\right)$ is not 2-MITEable.

- For the i^{th} row, if Λ_{ij} is $\pm 1/2$, then it means that j is in C and that there is a directed path from j to either $\beta(i)$ or $\gamma(i)$ but not both. From Theorem 4.3.1, it follows that $\beta(i)$ and $\gamma(i)$ belong to two different components.

- If the i^{th} row has a ± 2, then from Equation (4.9), it follows that $\beta(i)$ and $\gamma(i)$ belong to the same component.

- From the last two observations, it follows that the same row in Λ cannot have both a ± 2 and a $\pm 1/2$. Thus, $\Lambda = [1 \quad 1 \quad .5 \quad .5 \quad -2]$ is not 2-MITEable.

- If the i^{th} row does not have a $\pm 1/2$, then it means that $\beta(i)$ and $\gamma(i)$ both belong to the same component. For, if they belonged to different components, then $\Lambda_{ij} = \pm 1/2$ when j is in the nonempty directed circuits of the two components.

4.5 Single-Output POPL Networks

This research has not resulted in an optimal synthesis procedure for multiple-output systems of translinear equations using 2-MITE POPL networks. "Optimality" is used in the sense that there is no guarantee that the synthesis procedure to be presented in the next section produces a 2-MITE POPL network with the minimum possible number of MITEs. However, as will be shown in this section, a single translinear equation can be synthesized optimally using 2-MITEs. For the single-output case, the synthesis strategy to be followed here is as follows:

Problem Synthesize $I_{\mathrm{o}} = \prod_{j=1}^{n} I_j^{\Lambda_j}$, where $\sum_{j=1}^{m} \Lambda_j = 1$.

1. If Λ is 2-MITEable, then the problem is solved.

2. If Λ is not 2-MITEable, then use multiple copies of the input currents $\{I_j\}$ to produce a $\widetilde{\Lambda}$ that is 2-MITEable. In other words, each power Λ_j is split into different powers such that

$$\Lambda_j = \widetilde{\Lambda}_{t_{j-1}+1} + \widetilde{\Lambda}_{t_{j-1}+2} + \cdots + \widetilde{\Lambda}_{t_j} \tag{4.11}$$

   with $t_0 = 0$. The split is made such that the matrix $\widetilde{\Lambda}$ is 2-MITEable and minimum number of MITEs are used.

For this to work, we need three things:

1. Necessary and sufficient conditions for determining when Λ is 2-MITEable.

2. Procedure to synthesize the 2-MITE network(s) implementing Λ if it is 2-MITEable.

3. Procedure to determine the matrix $\widetilde{\Lambda}$ from Λ, in case Λ does not satisfy the above conditions.

4.5.1 Necessary and sufficient conditions for Λ to be 2-MITEable

We now discuss the conditions for a Λ matrix to be 2-MITEable for the single-output case i.e., Λ is a row vector. In the next section, we prove the following:

Theorem 4.5.1 *A $1 \times n$ vector Λ satisfying $\sum_j \Lambda_j = 1$ is 2-MITEable if and only if*

1. *$\Lambda_j \in \{0, 1/2, -1/2, 1, -1, 2, -2\}$ for every $j \in \{1, 2, \ldots, n\}$. However, Λ does not have both a ± 2 and a $\pm 1/2$.*

2. *If Λ does not have $\pm 1/2$, then the sum of the elements in Λ that are ± 2 is one of $\{+2, -2, 0\}$.*

3. *If Λ has $\pm 1/2$, then the sum of the elements in Λ that are $\pm 1/2$ is one of $\{+1, -1, 0\}$.*

Note:

The theorem can be stated in a simpler fashion when the powers are described by the *translinear loop matrix* A rather than Λ. It is easy to see that for the single output case, $A = [\Lambda \ \ -1]$. However, if we make sure that A has only integer powers, then $A = [2\Lambda \ \ -2]$ when Λ has $\pm 1/2$ and is $[\Lambda \ \ -1]$ otherwise. Then, the above theorem can be restated as : *A $1 \times (n+1)$ integer vector $A = [a_i]$ satisfying $\sum_j a_j = 0$ with no nonunity common divisor between its elements is 2-MITEable if and only if*

1. *$a_j \in \{0, 1, -1, 2, -2\}$ for every $j \in \{1, 2, \ldots, n+1\}$.*

2. *The sum of the elements in A that are ± 2 is one of $\{+2, -2, 0\}$.*

Let $y_j = \delta_{j\beta} + \delta_{j\gamma}$. From the observations made in the previous section, all the elements in the given Λ must belong to $\{0, 1/2, -1/2, 1, -1, 2, -2\}$. This allows only two cases that are discussed next.

4.5.2 Case when Λ has no powers that are $\pm 1/2$

In this case, β and γ are in the same component, with associated direct circuit C (say). Let $G = G_c(\widehat{X})$ be the Coates graph of $\widehat{X} = X - I_n$. From Theorem 4.3.1, we know that if C were contracted into a single vertex v, the resulting graph $\widetilde{G}$ is a rooted tree with v as the root. There are unique directed paths P_1 and P_2 from v to β and γ, respectively. Let δ be the "last" vertex in P_1 that is also in P_2 i.e., $\delta \in P_1 \cap P_2$ and $(\bar{\delta} P_1 \beta) \cap P_2 = \emptyset$. Note that this also means that $(\bar{\delta} P_2 \gamma) \cap P_1 = \emptyset$. Since there is only one directed path from v to δ, it follows that $v P_1 \delta = v P_2 \delta$.

$\boxed{\textbf{Case 1: } v \neq \delta}$

Here, $\widetilde{G}$ is of the form shown in Figure 4.8(a). Coming back to the original graph G, it is

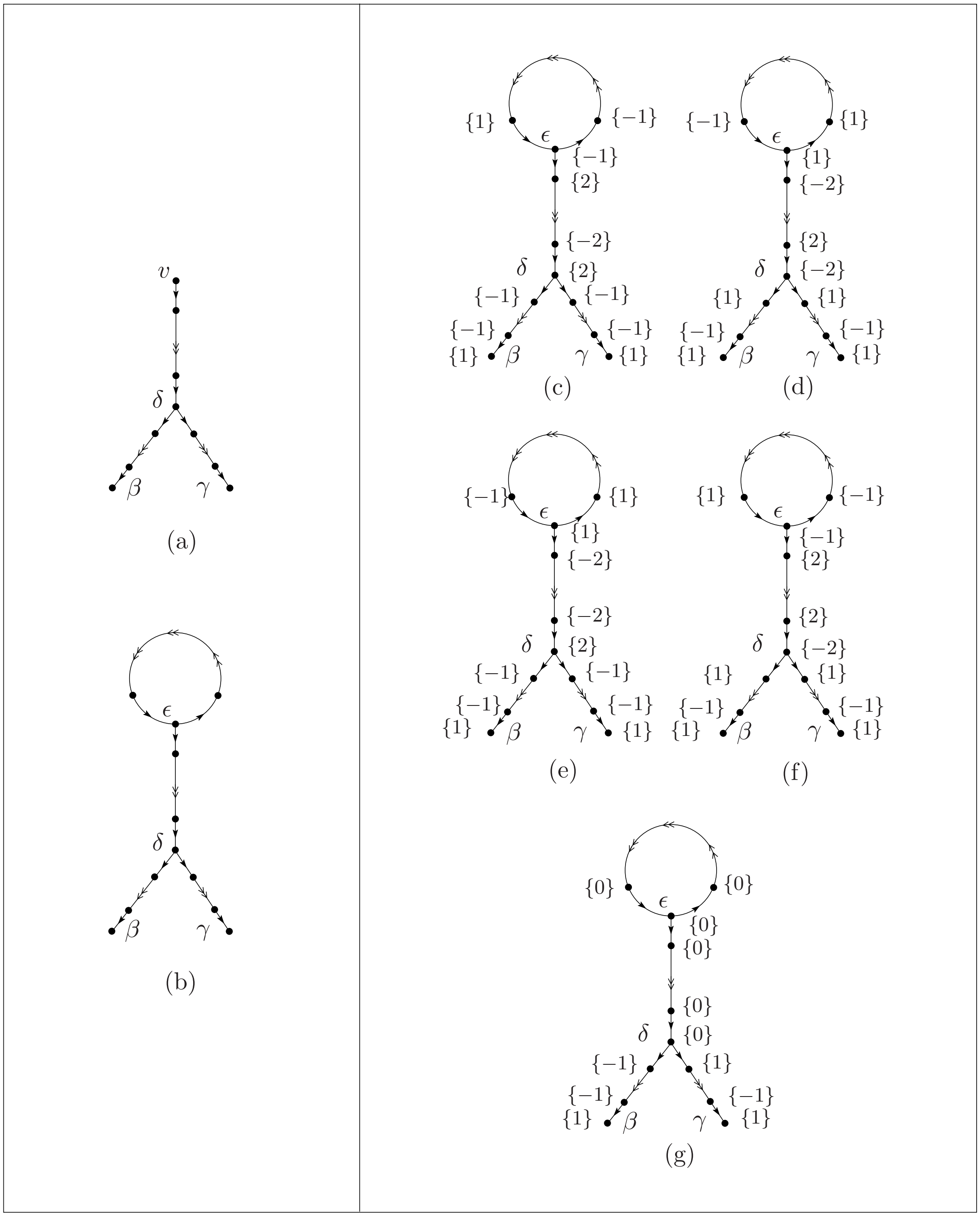


Figure 4.8. $G_c(\widehat{X})$ **for 2-MITE POPL networks with single outputs that has no** $\pm 1/2$ **powers and satisfies** $v \neq \delta$**. (a) is the graph** $\widetilde{G}$ **formed when the directed circuit in** $G_c(\widehat{X})$ **is contracted to a vertex** v **that is not equal to** δ**. (b) is the** $G_c(\widehat{X})$ **corresponding to the same** $\widetilde{G}$**. (c) is the case when the sum of** ± 2**-powers in** Λ **is** $+2$**. (d) is followed when the sum of** ± 2**-powers is** -2**. (e) is found when the sum of** ± 2**-powers is** 0 **and when** $\Lambda_\delta = +2$**. (f) is generated when the** ± 2**-powers add up to** 0 **and when** $\Lambda_\delta = -2$**. The cases (c)-(f) occur only** $d(\beta, \delta) - d(\gamma, \delta)$ **is even. In the odd case the powers occur as shown in (g). The sequence of** Λ_j **values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path.**

easy to see that there is a single vertex $\epsilon \in C$ that replaces v in the path to δ. Hence, the only possible Coates graph is of the form shown in Figure 4.8(b).

The path from ϵ to δ but excluding ϵ will be referred to as the *trunk* of G. The paths from δ to β and γ, but excluding δ, will be called the *limbs* of G. The values of Λ associated with the limbs, trunk, and the directed circuit C will be called the *powers in* the limbs, trunk, and C, respectively. It should be noted that while the limbs can be empty, the trunk cannot be empty in this case, since $v \neq \delta$.

$\boxed{\textbf{Case 1(a):}\ d(\beta,\delta) - d(\gamma,\delta)\ \textit{is even}}$

Clearly, in this case, $\Lambda_\delta = (-1)^{d(\beta,\delta)} + (-1)^{d(\gamma,\delta)} \neq 0$. Using Equation (4.9), we get:

$$\Lambda_j = \begin{cases} (-1)^{d(\gamma,j)} & \text{if } \delta \prec j \preceq \gamma \\ (-1)^{d(\beta,j)} & \text{if } \delta \prec j \preceq \beta \\ 2(-1)^{d(\beta,j)} & \text{if } \epsilon \prec j \preceq \delta \\ (-1)^{d(\beta,j)} & \text{if } j \in C \end{cases} \tag{4.12}$$

4.5.2.1 Synthesis

This is the only case where a ± 2 power is synthesized. It is easy to see that the sequence of ± 2 in the trunk alternate in sign. Let μ be the vertex after ϵ in the trunk. The sum of the different powers that are ± 2 is clearly $+2$ (when $\Lambda_\delta = +2$ and $\Lambda_\mu = +2$), or -2 (when $\Lambda_\delta = -2$ and $\Lambda_\mu = -2$), or 0 (when $\Lambda_\delta = \pm 2$ and $\Lambda_\mu = \mp 2$). It is clear that this satisfies the "only if" part of Theorem 4.5.1 for the case when no fractional powers are present. The "if" part is proved below by synthesizing appropriate MITE network(s) in each case:

$\boxed{\textit{1. The sum of } \pm 2 \textit{ powers in } \Lambda \textit{ is } +2}$

From the previous paragraph, this case requires $\Lambda_\delta = 2$ and $\Lambda_\mu = 2$, because the other configurations produce different sums of the ± 2 powers. The remaining ± 2 powers are arranged with alternating signs in the trunk. This case is depicted in Figure 4.8(c). Note that if there are s currents with power $+2$, then there must be $s-1$ currents with power -2 and hence there are $s!(s-1)!$ ways in which we can map the ± 2 currents to the vertices in the trunk.

The remaining powers, which are either 1 or -1 have to sum up to $1 - 2 = -1$. From

Equation (4.12), Λ_β and Λ_γ both are $+1$ and the first elements in the limbs have to be -1. Since the signs keeps alternating, the sum of the powers in the limbs has to be 0. Therefore, the sum of the powers in C is -1. It should also be noted that $\Lambda_\epsilon = -1$. Hence, once the powers in the trunk are fixed from the previous paragraph, the power of ϵ is fixed to be -1 and the remaining $\{+1, -1\}$ pairs are distributed as pairs on the limbs and the remaining parts of the directed circuit. Note that if there are k "$+1$"-powers and $k+1$ "-1"-powers, then the assignment of currents to vertices can be made in $(k+1)(k!)^2(k^2+3k+4)/4$ ways.

2. The sum of ± 2 powers in Λ is -2

Here, $\Lambda_\delta = -2$ and $\Lambda_\mu = -2$. The remaining powers follow as shown in Figure 4.8(d). Let there be s "-2"- powers and k "$+1$"-powers. It should be noted that $k \geq 3$ and that there are $s-1$ "$+2$"-powers and $k-3$ "-1"-powers. The synthesis is done as follows:

Step 1 Choose two currents that are raised to a power of -2 and assign them to vertices δ and μ. If there is only one -2-power, then $\delta = \mu$. The remaining ± 2 powers are arranged in the trunk with the signs alternating. This process can be done in $s!(s-1)!$ ways.

Step 2 Choose three currents with $+1$-powers and assign them to ϵ, β, and γ. The remaining $k-3$ pairs of $\{+1, -1\}$ are assigned as pairs in an alternating fashion to one or more of C and the limbs. This process can be done in $k((k-1)!)^2/4$ ways.

3. The sum of ± 2 powers in Λ is 0

Let there be s "$+2$"- powers and k "$+1$"-powers. It follows that there are s "-2"-powers and $k-1$ "-1"-powers. This case splits into two subcases:

3(a). $\Lambda_\delta = +2$

When the above choice is made, then it follows that $\Lambda_\mu = -2$. The synthesis is done as follows:

Step 1 Choose two currents that are raised to a power of $+2$ and -2 and assign them to vertices δ and μ,respectively. The remaining ± 2 powers are arranged in the trunk with the signs alternating. This process can be done in $(s!)^2$ ways.

Step 2 Choose a current with $+1$-power and assign it to ϵ. The remaining $k-1$ pairs of $\{+1, -1\}$ are assigned as pairs in an alternating fashion to one or more of C and the limbs. This process can be done in $k((k-1)!)^2(k^2+k+2)/4$ ways.

This case is depicted in Figure 4.8(e).

3(b). $\Lambda_\delta = -2$

Clearly, $\Lambda_\mu = +2$. To synthesize the circuit,

Step 1 Choose two currents that are raised to a power of -2 and $+2$ and assign them to vertices δ and μ,respectively. The remaining ± 2 powers are arranged in the trunk with the signs alternating. This process can be done in $(s!)^2$ ways.

Step 2 Choose a current with -1-power and assign it to ϵ. Choose two currents with $+1$-power and assign them to β and γ. The remaining $k-2$ pairs of $\{+1,-1\}$ are assigned to one or more of C and the limbs. This process can be done in $(k-1)(k!)^2/4$ ways.

This case is depicted in Figure 4.8(f).

It should be noted that both the above subcases produce MITE networks implementing the same equation. Hence, there are totally $(s!)^2\{(k-1)(k!)^2+k((k-1)!)^2(k^2+k+2)\}/4 = (s!(k-1)!)^2 k(k^2+1)/4$ ways of implementing this case.

Case 1(b): $d(\beta,\delta) - d(\gamma,\delta)$ *is odd*

In this case, $\Lambda_\delta = (-1)^{d(\beta,\delta)} + (-1)^{d(\gamma,\delta)} = 0$. Using Equation (4.9), we get:

$$\Lambda_j = \begin{cases} (-1)^{d(\gamma,j)} & \text{if } \delta \prec j \preceq \gamma \\ (-1)^{d(\beta,j)} & \text{if } \delta \prec j \preceq \beta \\ 0 & \text{if } j \preceq \delta \end{cases} \tag{4.13}$$

This case is shown in Figure 4.8(g). It is clear that no ± 2 powers are generated here. Note that some of the powers are 0. For single-output networks, it might be argued that 0-powers are not needed. However, in case one is synthesizing multiple-output POPL networks by synthesizing each equation separately and consolidating whenever possible, then these redundancies are of immense help.

Let there be s "-1"-powers, implying that there are $s+1$ "$+1$"-powers. For synthesis, the extra "$+1$"-power is first placed in one of the limbs and the remaining $\{+1,-1\}$ pairs are then distributed among either limbs with the signs alternating. This can be done in

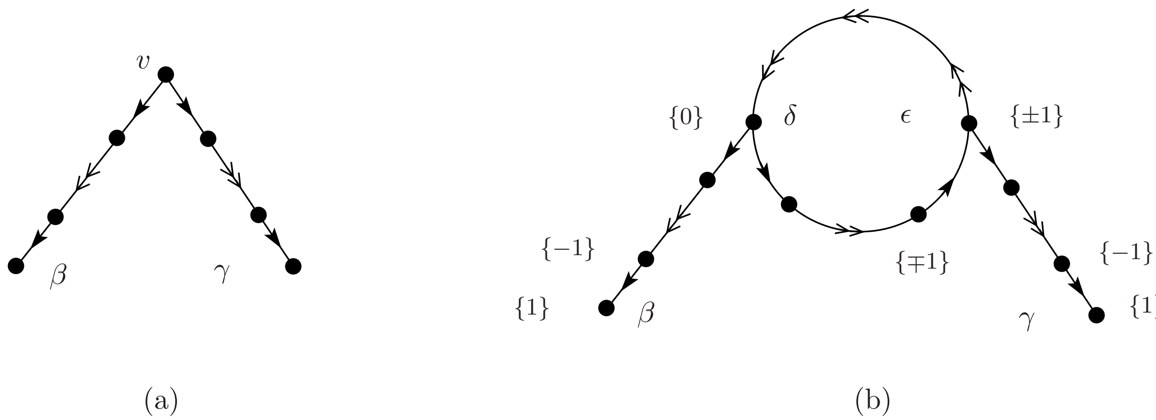


Figure 4.9. $G_c(\widehat{X})$ for 2-MITE POPL networks with single outputs that has no $\pm 1/2$ powers and satisfies $v = \delta$. (a) is the graph $\widetilde{G}$ formed when the directed circuit in $G_c(\widehat{X})$ is contracted to a vertex v. (b) is the $G_c(\widehat{X})$ corresponding to the same $\widetilde{G}$. Without loss of generality, $\Lambda_\epsilon \neq 0$ has been assumed. The sequence of Λ_j values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path.

$((s+1)!)^2$ ways.

$\boxed{\textbf{Case 2: } v = \delta}$

It should be noted that here, there might be two vertices δ and ϵ, not necessarily distinct, such that the paths from v to β (v to γ) in $\widetilde{G}$ corresponds to paths from δ to β (ϵ to γ) in $G_c(\widehat{X})$, as shown in Figure 4.9. Let k be the odd length of the directed circuit. In this case,

$$\begin{aligned}
\Lambda_\epsilon &= \frac{1}{2}\left((-1)^{d(\gamma,\epsilon)} + (-1)^{d(\beta,\epsilon)}\right) \\
&= \frac{1}{2}\left((-1)^{d(\gamma,\epsilon)} + (-1)^{d(\beta,\delta)+d(\delta,\epsilon)}\right) \\
&= \frac{1}{2}\left((-1)^{d(\gamma,\epsilon)} + (-1)^{d(\beta,\delta)+k-d(\epsilon,\delta)}\right) \\
&= \frac{1}{2}(-1)^{d(\gamma,\epsilon)}\left(1 - (-1)^{d(\beta,\delta)+d(\epsilon,\delta)-d(\gamma,\epsilon)}\right)
\end{aligned} \tag{4.14}$$

Similarly, $\Lambda_\delta = 1/2(-1)^{d(\beta,\delta)}\left(1 + (-1)^{d(\beta,\delta)+d(\epsilon,\delta)-d(\gamma,\epsilon)}\right)$. From this, it is clear that only one of Λ_δ or Λ_ϵ is nonzero. Without loss of generality, we will assume $\Lambda_\epsilon \neq 0$, i.e., $d(\beta,\delta) + d(\epsilon,\delta) - d(\gamma,\epsilon)$ is odd. The remaining powers are as follows:

$$\Lambda_j = \begin{cases}
(-1)^{d(\beta,j)} & \text{if } \delta \prec j \preceq \beta \\
(-1)^{d(\gamma,j)} & \text{if } \epsilon \prec j \preceq \gamma \\
(-1)^{d(\gamma,j)} & \text{if } \delta \prec j \preceq \epsilon \\
0 & \text{if } \epsilon \prec j \preceq \delta
\end{cases} \tag{4.15}$$

4.5.3 Case when Λ has some powers that are $\pm 1/2$

In this case, β and γ are in different components, with associated directed circuits C_1 and C_2. The only possible graph structure is as shown in Figure 4.10(a), allowing for $\beta = \delta$ or $\gamma = \epsilon$. It is a easy consequence of Equation (4.9) that

$$\Lambda_j = \begin{cases} (-1)^{d(\beta,j)} & \text{if } \delta \prec j \preceq \beta \\ (-1)^{d(\beta,j)}/2 & \text{if } j \in C_1 \\ (-1)^{d(\gamma,j)} & \text{if } \epsilon \prec j \preceq \gamma \\ (-1)^{d(\gamma,j)}/2 & \text{if } j \in C_2 \end{cases} \tag{4.16}$$

4.5.3.1 Synthesis

1. The sum of $\pm 1/2$ powers in Λ is 1

In this case, $\Lambda_\delta = \Lambda_\epsilon = 1/2$. If there are k "$+1/2$" powers and s "$+1$" powers, it follows that there are $k - 2$ "$-1/2$" powers and s "-1" powers. For the synthesis, p "$+1$" powers, p "-1" powers, q "$+1/2$" powers, and $q - 1$ "$-1/2$" powers are chosen, with $0 \leq p \leq s$ and $1 \leq q \leq k - 1$, in order that one of the components is formed. The other component is formed from the remaining powers. This results in $(s + 1)(s!)^2 k((k - 1)!)^2/2$ possible 2-MITE networks. This case is shown in Figure 4.10(b).

2. The sum of $\pm 1/2$ powers in Λ is -1

In this case, $\Lambda_\delta = \Lambda_\epsilon = -1/2$. If there are k "$+1/2$" powers and s "$+1$" powers, it follows that there are $k + 2$ "$-1/2$" powers and $s - 2$ "-1" powers. The synthesis is done similar to the previous case. This results in $s((s-1)!)^2(k+2)((k+1)!)^2/2$ 2-MITE networks. This case is shown in Figure 4.10(c).

3. The sum of $\pm 1/2$ powers in Λ is 0

In this case, $\Lambda_\delta = +1/2$ and $\Lambda_\epsilon = -1/2$. If there are k "$+1/2$" powers and s "$+1$" powers, it follows that there are k "$-1/2$" powers and $s - 1$ "-1" powers. This results in $(s!)^2 k((k - 1)!)^2$ 2-MITE networks. This case is shown in Figure 4.10(d).

4.6 2-MITE synthesis of arbitrary POPL equations with a single output

Summary: This section answers the question: Given a single-output POPL equation, how do we go about implementing it using 2-MITEs without using more MITEs than we

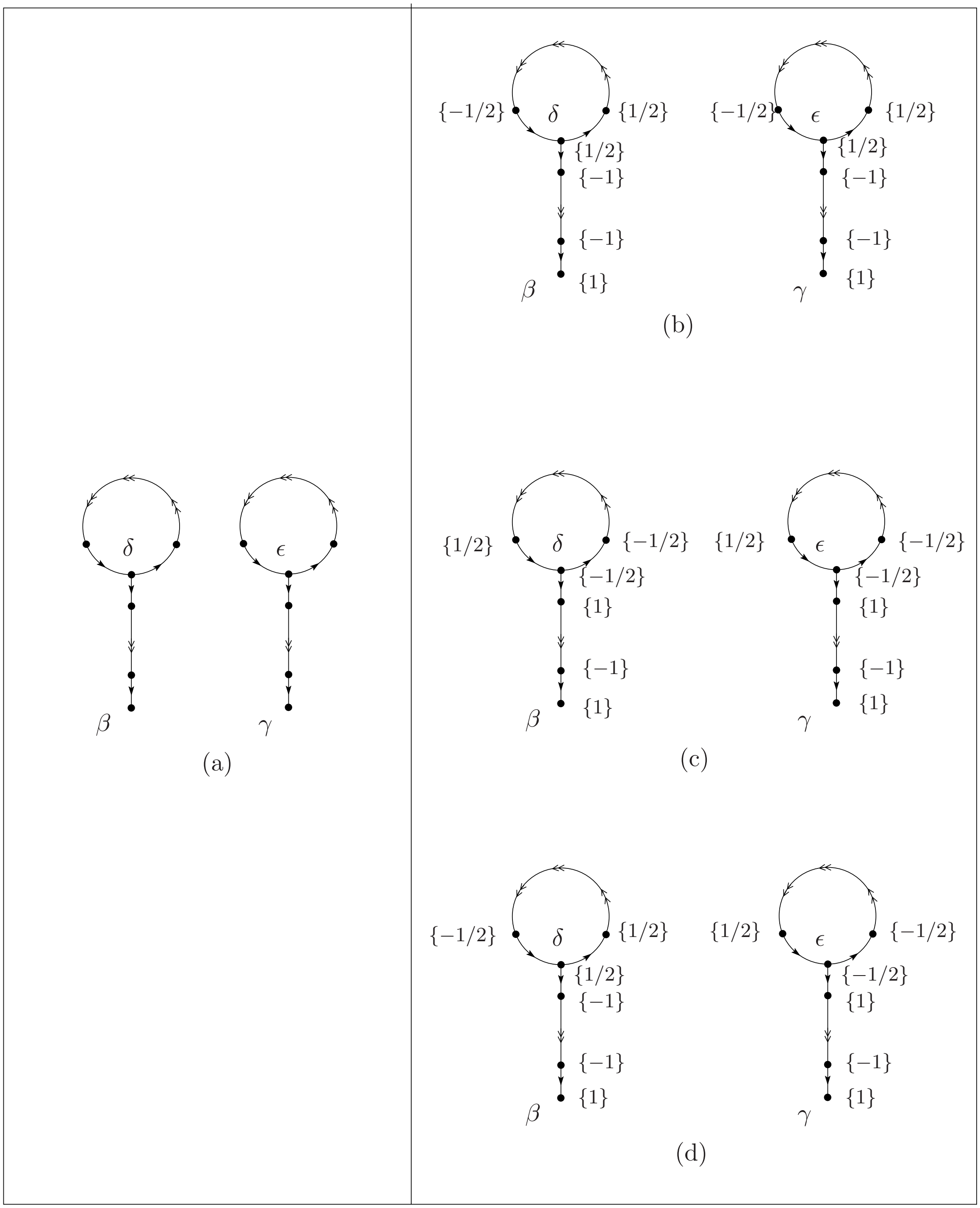


Figure 4.10. $G_c(\widehat{X})$ **for 2-MITE POPL networks with single outputs that has some $\pm 1/2$ powers. (a) is the general form of $G_c(\widehat{X})$ required when $\pm 1/2$ powers are present. (b) is the case when the sum of $\pm 1/2$-powers in Λ is $+1$. (c) is followed when the sum of $\pm 1/2$-powers is -1. (d) is found when the sum of $\pm 1/2$-powers is 0. The sequence of Λ_j values are shown for each section. The double arrows indicate a sequence of directed edges forming a directed path.**

need to? In other words, we (the users) have decided to not trade the fan–in of a network in order to keep the number of MITEs fixed, as was done in Chapter 3, but instead are ready to use more *number* of MITEs and copies of currents to do the same job. However, we are also not willing to use more than the optimal number of MITEs needed to implement the equation.

It should be noted that the *existence* of a 2-MITE network implementing $I_o = \prod_{k=1}^{n} I_k^{\Lambda_k}$ can be proved using methods developed prior to this thesis, as discussed in Chapter 1. Here, we will use the development of the theory of 2-MITE networks to *optimize* the network for minimum number of MITEs.

First, we demonstrate this for the case when all the powers Λ_k are integers and then move on to the general case. In this special case, we note the following:

- Since only those powers that are $\pm 2, \pm 1$, and 0 are allowed for 2-MITE synthesis, we need to additively split Λ_k into sums of these allowed powers. In other words, we need to find, for each k, the quadruplet of nonnegative integers : $(p_1(k), p_{-1}(k), p_2(k), p_{-2}(k))$ such that $\Lambda_k = (+1)p_1(k) + (-1)p_{-1}(k) + (+2)p_2(k) + (-2)p_{-2}(k)$. Here $p_i(k)$ is the number of times i is present when Λ_k is split as a sum of $\pm 1, \pm 2$.

- Since we are trying to reduce the number of MITEs needed in this *single-output* synthesis, note that a minimal solution is only going to have either $p_i(k) \geq 0$ with $p_{-i}(k) = 0$ or vice versa. For example, having $\cdots I_j^2 I_j^{-2} \cdots$ needs more MITEs than $\cdots I_j^0 \cdots$.

- Clearly, it is enough to do the splitting as $\Lambda = 2\mathbf{u} + \mathbf{v}$, where $\mathbf{u}, \mathbf{v}$ are constrained to be vectors of integers, though not necessarily nonnegative integers. For example, if $u_k > 0$, then we have the product I_k^2 occurring u_k times.

- For 2-MITEability, we require $\sum_k u_k \in \{+1, -1, 0\}$.

- The number of MITEs needed for the implementation is clearly $\sum_{k=1}^{n} |u_k| + |v_k|$. This the *objective function* that we want to minimize.

Hence, the optimal 2-MITE synthesis problem can be written mathematically as follows:

Problem:

Given $\Lambda = [\Lambda_k] \in \mathcal{M}_{1,n}(\mathbb{N})$ satisfying $\Lambda \mathbf{1}_n = 1$, minimize the *objective function* $f(\mathbf{u}, \mathbf{v}) = \|\mathbf{u}\|_1 + \|\mathbf{v}\|_1 = \sum_k (|u_k| + |v_k|)$, where $\mathbf{u} = [u_k], \mathbf{v} = [v_k] \in \mathcal{M}_{1,n}(\mathbb{Z})$, subject to the *constraints*

1. $\Lambda_k = 2u_k + v_k$ for all $i \in [1:n]$
2. $\sum_{k=1}^{n} u_k \in \{+1, -1, 0\}$

We will call $(\mathbf{u}^\star, \mathbf{v}^\star) \in \mathcal{M}_{1,2n}(\mathbb{Z})$ a *minimal solution of* Λ if for all $(\mathbf{u}, \mathbf{v}) \in \mathcal{M}_{1,2n}(\mathbb{Z})$ satisfying the constraints, $f(\mathbf{u}^\star, \mathbf{v}^\star) \leq f(\mathbf{u}, \mathbf{v})$.

We will develop the solution through a series of lemmas:

Lemma 4.6.1 *If* $(\mathbf{u}, \mathbf{v})$ *is a minimal solution of* Λ*, then* $u_k \Lambda_k \geq 0$*. In other words,* u_k *either has the same sign as* Λ_k *or is zero.*

Proof : Suppose the lemma is not true and that there is a $s \in [1:n]$ such that $u_s \Lambda_s < 0$. We will assume that $\Lambda_s > 0$; the proof for the case $\Lambda_s < 0$ is similar. Since $\Lambda_s - 2u_s = v_s$ and $u_s \leq -1$, it follows that $v_s > 2$. Now, it is clear that $\sum_{k \neq s} u_k + u_s = 1$ since otherwise $(\mathbf{u}^\star, \mathbf{v}^\star)$ defined by

$$u_k^\star = \begin{cases} u_k & \text{if } k \neq s \\ u_s + 1 & k = s \end{cases} \qquad v_k^\star = \begin{cases} v_k & \text{if } k \neq s \\ v_s - 2 & k = s \end{cases} \tag{4.17}$$

satisfies the constraints and

$$\begin{aligned} f(\mathbf{u}^\star, \mathbf{v}^\star) &= f(\mathbf{u}, \mathbf{v}) - |u_s| + |u_s + 1| - |v_s| + |v_s - 2| \\ &= f(\mathbf{u}, \mathbf{v}) + u_s - (u_s + 1) - v_s + (v_s - 2) \\ &= f(\mathbf{u}, \mathbf{v}) - 3, \end{aligned} \tag{4.18}$$

which contradicts optimality of $(\mathbf{u}, \mathbf{v})$. Since $\sum u_k = 1$, there is some $t \in [1:n]$ such that $u_t \geq 1$. It is then clear that $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ defined by

$$\tilde{u}_k = \begin{cases} u_k & \text{if } k \neq s, t \\ u_s + 1 & k = s \\ u_t - 1 & k = t \end{cases} \qquad \tilde{v}_k = \begin{cases} v_k & \text{if } k \neq s, t \\ v_s - 2 & k = s \\ v_t + 2 & k = t \end{cases} \tag{4.19}$$

satisfies the constraints and

$$\begin{aligned} f(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}) &= f(\mathbf{u}, \mathbf{v}) - |u_s| + |u_s + 1| - |u_t| + |u_t - 1| - |v_s| + |v_s - 2| - |v_t| + |v_t + 2| \\ &= f(\mathbf{u}, \mathbf{v}) + u_s - (u_s + 1) - u_t + (u_t - 1) - v_s + (v_s - 2) - |v_t| + |v_t + 2| \\ &\leq f(\mathbf{u}, \mathbf{v}) - 4 - |v_t| + |v_t| + 2 \\ &\leq f(\mathbf{u}, \mathbf{v}) - 2 \end{aligned} \tag{4.20}$$

which contradicts the assumption that the lemma is not true. $\square$

Lemma 4.6.2 *If* $(\mathbf{u}, \mathbf{v})$ *is a minimal solution of* Λ*, then* $v_k \geq -1$ *if* $\Lambda_k > 0$ *and* $v_k \leq 1$ *in case* $\Lambda_k < 0$*. In other words,* v_k *either has the same sign as* Λ_k *or is* 0 *or is* $-\operatorname{sign} \Lambda_k$*.*

Proof : Let there be a $s \in [1 : n]$ such that, contrary to the claim, $v_s \leq -2$ when $\Lambda_s > 0$. We will show that this leads to a contradiction and leave the case where there is a $v_s \geq 2$ with $\Lambda_s < 0$ to the reader. We have $2u_s = \Lambda_s - v_s > 2$, i.e., $u_s > 1$. It follows that $\sum_k u_k = -1$, for otherwise $(\mathbf{u}^\star, \mathbf{v}^\star)$ defined by

$$u_k^\star = \begin{cases} u_k & \text{if } k \neq s \\ u_s - 1 & k = s \end{cases} \qquad v_k^\star = \begin{cases} v_k & \text{if } k \neq s \\ v_s + 2 & k = s \end{cases} \tag{4.21}$$

satisfies the constraints and

$$\begin{aligned} f(\mathbf{u}^\star, \mathbf{v}^\star) &= f(\mathbf{u}, \mathbf{v}) - |u_s| + |u_s - 1| - |v_s| + |v_s + 2| \\ &= f(\mathbf{u}, \mathbf{v}) - u_s + (u_s - 1) + v_s - (v_s + 2) \\ &= f(\mathbf{u}, \mathbf{v}) - 3, \end{aligned} \tag{4.22}$$

which contradicts optimality of $(\mathbf{u}, \mathbf{v})$. Since $\sum_k u_k = -1$, there is a $t \in [1 : n]$ with $u_t \leq -1$. It is then clear that $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ defined by

$$\tilde{u}_k = \begin{cases} u_k & \text{if } k \neq s, t \\ u_s - 1 & k = s \\ u_t + 1 & k = t \end{cases} \qquad \tilde{v}_k = \begin{cases} v_k & \text{if } k \neq s, t \\ v_s + 2 & k = s \\ v_t - 2 & k = t \end{cases} \tag{4.23}$$

satisfies the constraints and

$$
\begin{aligned}
f(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}) &= f(\mathbf{u}, \mathbf{v}) - |u_s| + |u_s - 1| - |u_t| + |u_t + 1| - |v_s| + |v_s + 2| - |v_t| + |v_t - 2| \\
&= f(\mathbf{u}, \mathbf{v}) - u_s + (u_s - 1) + u_t - (u_t + 1) + v_s - (v_s + 2) - |v_t| + |v_t - 2| \\
&\leq f(\mathbf{u}, \mathbf{v}) - 4 - |v_t| + |v_t| + 2 \\
&\leq f(\mathbf{u}, \mathbf{v}) - 2
\end{aligned}
\tag{4.24}
$$

which is a contradiction. □

Lemma 4.6.3 *If* $\Lambda_s \geq 2$ *and* $\Lambda_t \leq -2$ *for some* $s, t \in [1 : n]$*, then there is a minimal solution* $(\mathbf{u}^\star, \mathbf{v}^\star)$ *with* $u_s^\star \geq 1$ *and* $u_t^\star \leq -1$.

Proof : By Lemmas 4.6.1 and 4.6.2, any minimal solution $(\mathbf{u}, \mathbf{v})$ should have $u_s > 0 > u_t$. We prove the lemma by proving three statements:

Step 1: No minimal solution $(\mathbf{u}, \mathbf{v})$ has $u_s = 0 = u_t$.

Step 2: If a minimal solution $(\mathbf{u}, \mathbf{v})$ has $u_s = 0$ and $u_t \leq -1$, then there is a minimal solution $(\mathbf{u}^\star, \mathbf{v}^\star)$ with $u_s^\star \geq 1$ and $u_t^\star \leq -1$.

Step 3: If a minimal solution $(\mathbf{u}, \mathbf{v})$ has $u_s \geq 1$ and $u_t = 0$, then there is a minimal solution$(\mathbf{u}^\star, \mathbf{v}^\star)$ with $u_s^\star \geq 1$ and $u_t^\star \leq -1$.

Step 1: If $(\mathbf{u}, \mathbf{v})$ is a minimal solution with $u_s = u_t = 0$, then $v_s = \Lambda_s \geq 2$ and $v_t = \Lambda_t \leq -2$. It is easily verified that $(\tilde{u}, \tilde{v})$ defined by

$$
\tilde{u}_k = \begin{cases} u_k & \text{if } k \neq s, t \\ 1 & k = s \\ -1 & k = t \end{cases} \qquad \tilde{v}_k = \begin{cases} v_k & \text{if } k \neq s, t \\ v_s - 2 & k = s \\ v_t + 2 & k = t \end{cases} \tag{4.25}
$$

provides a contradiction to $(\mathbf{u}, \mathbf{v})$ being a minimal solution.

Step 2: If a minimal solution $(\mathbf{u}, \mathbf{v})$ has $u_s = 0$ and $u_t \leq -1$, then it follows that $v_s \geq 2$. By reasoning similar to the previous two lemmas, it follows that $\sum_k u_k = 1$, implying that there is some $j \in [1:n]$ with $u_j \geq 1$. Then, $(\mathbf{u}^\star, \mathbf{v}^\star)$ defined by

$$u_k^\star = \begin{cases} u_k & \text{if } k \neq s, j \\ 1 & k = s \\ u_j - 1 & k = j \end{cases} \qquad v_k^\star = \begin{cases} v_k & \text{if } k \neq s, j \\ v_s - 2 & k = s \\ v_j + 2 & k = j \end{cases} \tag{4.26}$$

satisfies the constraints and

$$\begin{aligned} f(\mathbf{u}^\star, \mathbf{v}^\star) &= f(\mathbf{u}, \mathbf{v}) - |0| + |1| - |u_j| + |u_j - 1| - |v_s| + |v_s - 2| - |v_j| + |v_j + 2| \\ &= f(\mathbf{u}, \mathbf{v}) + 1 - u_j + (u_j - 1) - v_s + (v_s - 2) - |v_j| + |v_j + 2| \\ &= f(\mathbf{u}, \mathbf{v}) - 2 - |v_j| + |v_j + 2| \end{aligned} \tag{4.27}$$

By the previous lemma, $v_j \geq -1$, since $\Lambda_j > 0$ as $u_j \geq 1$. If $v_j \geq 0$, then $f(\mathbf{u}^\star, \mathbf{v}^\star) = f(\mathbf{u}, \mathbf{v}) - 2 + 2 = f(\mathbf{u}, \mathbf{v})$, which proves the result and if $v_j = -1$, $f(\mathbf{u}^\star, \mathbf{v}^\star) = f(\mathbf{u}, \mathbf{v}) - 2 - 1 + 1 = f(\mathbf{u}, \mathbf{v}) - 2$, which contradicts the hypothesis that $(\mathbf{u}, \mathbf{v})$ is a minimal solution.

Step 3: This closely follows the proof in Step 2 and hence will be left to the reader. □

Lemma 4.6.4 *If $\Lambda_s \geq 2$ and $\Lambda_t \leq -2$ for some $s, t \in [1:n]$, let*

$$\widetilde{\Lambda}_k = \begin{cases} \Lambda_k & \text{if } k \neq s, t \\ \Lambda_s - 2 & \text{if } k = s \\ \Lambda_t + 2 & \text{if } k = t \end{cases} \tag{4.28}$$

If $(\tilde{\mathbf{u}} = [\tilde{u}_k], \tilde{\mathbf{v}} = [\tilde{v}_k])$ is a minimal solution for $\widetilde{\Lambda}$, then $(\mathbf{u}^\star = [u_k^\star], \mathbf{v}^\star = [v_k^\star])$ is a minimal solution for Λ, where $\mathbf{v}^\star = \tilde{\mathbf{v}}$ and

$$u_k^\star = \begin{cases} \tilde{u}_k & \text{if } k \neq s, t \\ \tilde{u}_s + 1 & \text{if } k = s \\ \tilde{u}_t - 1 & \text{if } k = t \end{cases} \tag{4.29}$$

Proof : Since $\widetilde{\Lambda}_s \geq 0$ and $\widetilde{\Lambda}_t \leq 0$, by Lemma 4.6.1, $\tilde{u}_s \geq 0$ and $\tilde{u}_t \leq 0$. Hence,

$$\begin{aligned} f(\mathbf{u}^\star, \mathbf{v}^\star) &= f(\tilde{\mathbf{u}}, \tilde{\mathbf{v}}) - |\tilde{u}_s| + |\tilde{u}_s + 1| - |\tilde{u}_t| + |\tilde{u}_t - 1| \\ &= f(\tilde{u}, \tilde{v}) - \tilde{u}_s + \tilde{u}_s + 1 + \tilde{u}_t - (\tilde{u}_t - 1) \\ &= f(\tilde{u}, \tilde{v}) + 2 \end{aligned} \tag{4.30}$$

Suppose $(\mathbf{u}^\star, \mathbf{v}^\star)$ is not a minimal solution for Λ, then there is a minimal solution $(\mathbf{u}, \mathbf{v})$ such that $f(\mathbf{u}, \mathbf{v}) < f(\mathbf{u}^\star, \mathbf{v}^\star)$. From Lemma 4.6.3, we can assume that this minimal solution satisfies $u_s \geq 1$ and $u_t \leq -1$. Now we construct a new solution, maybe not a minimal one, $(\hat{\mathbf{u}}, \hat{\mathbf{v}})$ to $\widetilde{\Lambda}$ as follows:

$$\hat{u}_k = \begin{cases} u_k & \text{if } k \neq s, t \\ u_s - 1 & \text{if } k = s \\ u_t + 1 & \text{if } k = t \end{cases} \qquad \hat{\mathbf{v}} = \mathbf{v} \tag{4.31}$$

It is easily verified that $(\hat{\mathbf{u}}, \hat{\mathbf{v}})$ satisfies the constraints with respect to $\widetilde{\Lambda}$. Further, $f(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = f(\mathbf{u}, \mathbf{v}) - 2$. Thus, we have $f(\hat{\mathbf{u}}, \hat{\mathbf{v}}) = f(\mathbf{u}, \mathbf{v}) - 2 < f(\mathbf{u}^\star, \mathbf{v}^\star) - 2 = f(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ which contradicts the assumption that $(\tilde{\mathbf{u}}, \tilde{\mathbf{v}})$ is a minimal solution to $\widetilde{\Lambda}$. $\square$

Let us look at what Lemma 4.6.4 says through an example. Let $\Lambda = [5 \quad -3 \quad -1]$ i.e., we want to implement $I_4 = I_1^5 I_2^{-3} I_3^{-1}$ using 2-MITEs. Lemma 4.6.4 tells us that to find the 2-MITE network with minimal number of MITEs implementing this equation, it suffices to find the minimal solution of $I_1^3 I_2^{-1} I_3^{-1}$. We are effectively *extracting* out $I_1^2 I_2^{-2}$ out of the equation for I_4. Let us write this as $I_4 = (I_1^2 I_2^{-2})(I_1^3 I_2^{-1} I_3^{-1})$. The product in the second parentheses can be written as $I_1^2 I_1 I_2^{-1} I_3^{-1}$ which is clear 2-MITEable. Notice that this *extraction* process is itself highly intuitive and almost obvious. One might then ask, what is the use of going through all these lemmas? The answer is that while intuition *guides* us to the solution, the proofs are needed to *establish that the intuition is correct.* It should be noted that a cascade network synthesis of this equation would have required 10 MITEs, including the output one, as opposed to the 7 MITEs required in this synthesis. Clearly, the difference increases as the number of $+2, -2$ pairs we can "extract" increases.

Once the extraction process is complete, there are only three possibilities:

1. $\Lambda_k \in \{-1, 0, 1\}$ for each $k \in [1 : n]$. The solution is then straightforward, given that it is clear 2-MITEable and simply uses the synthesis methods described in previous sections in this chapter. Clearly, $\|\Lambda\|_1$ input MITEs are needed for the solution.

2. There is a $\Lambda_s \geq 2$. Clearly, if $\Lambda_t < 0$, $\Lambda_t = -1$, since otherwise the extraction would not be complete. In this case, we can extract out a lone $+2$ with the remaining powers

being ± 1, which keeps Λ 2-MITEable. Hence, the number of input MITEs needed in this case is $\|\Lambda\|_1 - 1$. If, after extracting the $+2$, there is still some $\Lambda_j \geq 2$, then since there should be a $\Lambda_t = -1$, we can implement the product either as $\cdots I_j^1 I_j^1 I_t^{-1} \cdots$ or as $\cdots I_j^2 I_t^{-2} I_t^1$. Note that the number of MITEs remains the same and hence each is a minimal solution. For example, $I_1^4 I_2^{-1} I_3^{-1} I_4^{-1}$ can be split minimally as one of $I_1^2 I_1 I_1 I_2^{-1} I_3^{-1} I_4^{-1}$ $I_1^2 I_1^2 I_2^{-2} I_2^1 I_3^{-1} I_4^{-1}$, $I_1^2 I_1^2 I_2^{-1} I_3^{-2} I_3^1 I_4^{-1}$, $I_1^2 I_1^2 I_2^{-1} I_3^{-1} I_4^{-2} I_4^1$.

3. There is a $\Lambda_t \leq -2$. If $\Lambda_s > 0$, $\Lambda_s = 1$. In this case, a -2 is extracted out, the implementation taking $\|\Lambda\|_1 - 1$ input MITEs. Similar to the previous case, $\cdots I_j^{-2} I_s^1 \cdots$ is implemented either as $\cdots I_j^{-1} I_j^{-1} I_s^1 \cdots$ or as $\cdots I_j^{-2} I_s^2 I_s^{-1} \cdots$.

The following algorithm for constructing a minimal solution(s) to a given Λ follows:

Step 1: Initialize $\mathbf{u} = 0$, $\mathbf{v} = \Lambda$.

Step 2: Choose $v_s \geq 2$, $v_t \leq -2$. Replace $u_s \mapsto u_s + 1, v_s \mapsto v_s - 2, u_t \mapsto u_t - 1, v_t \mapsto v_t + 2$. If no such s, t exist, go to Step 4.

Step 3: Go to Step 2.

Step 4: If there is no $v_s \geq 2$ or a $v_t \leq -2$, solution process is complete.

Step 5: If there is a $v_s \geq 2$, replace $u_s \mapsto u_s + 1, v_s \mapsto v_s - 2$. If, after this, there is some $v_j \geq 2$, then *optionally* replace $u_j \mapsto u_j + 1, v_j \mapsto v_j - 2, u_t \mapsto u_t - 1, v_t \mapsto v_t + 2$ where t is chosen so that $v_t = -1$ before the replacement.

Step 6: If there is a $v_t \leq -2$, replace $u_t \mapsto u_t - 1, v_t \mapsto v_t + 2$. If, after this, there is some $v_j \leq -2$, then *optionally* replace $u_j \mapsto u_j - 1, v_j \mapsto v_j + 2, u_s \mapsto u_s + 1, v_s \mapsto v_s - 2$ where s is chosen so that $v_s = 1$ before the replacement.

Step 7: Go to Step 4.

Once a minimal solution $(\mathbf{u}, \mathbf{v})$ is found with minimum number of input MITEs given by $f(\mathbf{u}, \mathbf{v})$, to implement this in terms of 2-MITEs, the following is done:

Step 1: A 2-MITE network consisting of $f(\mathbf{u}, \mathbf{v})$ input MITEs and 1 output MITE is drawn.

Step 2: For each $k \in [1 : n]$, $|u_k| + |v_k|$ copies of the input current I_k is made and is fed to the same number of MITEs.

Step 3: A 2-MITEable POPL equation that is equivalent of Λ is obtained by replacing

$I_k^{\Lambda_k}$ in $\cdots I_k^{\Lambda_k} \cdots$ by

$$\begin{array}{ll} (\prod_{i=1}^{u_k} I_k^2)(\prod_{i=1}^{v_k} I_k) & \text{if } \Lambda_k > 0, v_k \geq 0 \\ (\prod_{i=1}^{u_k} I_k^2)\ I_k^{-1} & \text{if } \Lambda_k > 0, v_k = -1 \\ (\prod_{i=1}^{|u_k|} I_k^{-2})(\prod_{i=1}^{|v_k|} I_k^{-1}) & \text{if } \Lambda_k < 0, v_k \leq 0 \\ (\prod_{i=1}^{|u_k|} I_k^{-2})\ I_k & \text{if } \Lambda_k < 0, v_k = 1 \end{array}$$

Step 4: The 2-MITEable POPL equation is synthesized using the methods discussed in previous sections or using the method of diophantine equations in Chapter 3.

4.6.1 General case: Rational power matrix

We now consider the case when the elements of the given power matrix Λ are rational numbers, not simply integers alone. Here, it is found advantageous to move to the translinear loop matrix representation of $A = [\Lambda \ \ -1]$. Note that A can be multiplied by any number without changing the implemented function itself and so we multiply A by the least common multiple of the positive denominators of the coefficients of A. Hence, we can consider A to be composed of integers and can also assume that the elements in A have no common divisor except unity. The condition for 2-MITE implementation of A is that the coefficients should only be $\pm 2, \pm 1, 0$ and the ± 2 coefficients in A should add up to $\{+2, 0, -2\}$, as was discussed before.

It is easily seen that the "$+2, -2$" pair extraction method is straight away applicable here with very minor changes to account for $\Lambda \mathbf{1}_n = 1$ versus $A \mathbf{1}_{n+1} = 0$. The main difference arises in the *implementation* of the 2-MITE network and not the *optimization* itself. Since a_{n+1} is associated with the output current, which is not known a priori, the copies of the output current for feeding as inputs into the MITES is derived from the output MITE itself. To clarify, we will simply demonstrate the differences using an example:

Example: **Problem:** Synthesize $\Lambda = [1/4 \ \ 3/4]$

Here $A = [1 \ \ 3 \ \ -4]$, after multiplying $[\Lambda \ \ -1]$ by 4. Writing in terms of the currents, we

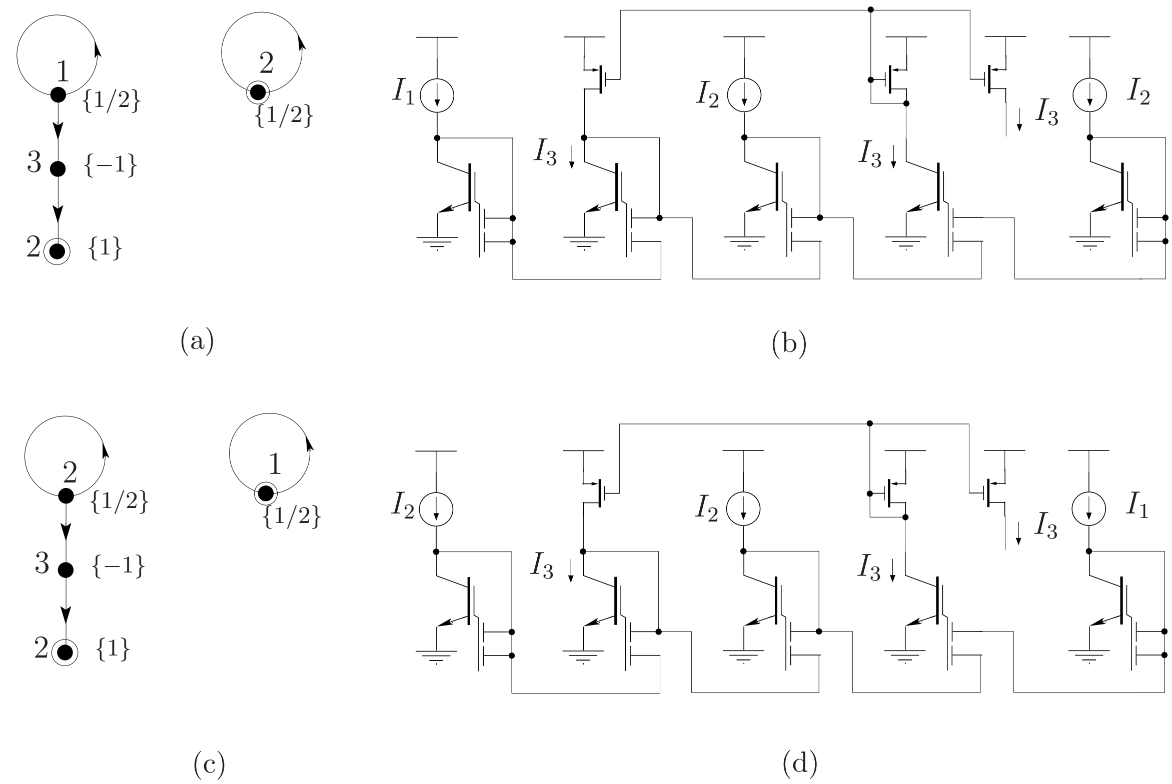


Figure 4.11. Optimal 2-MITE synthesis of the equation $I_3 = I_1^{1/4} I_2^{3/4}$. The only minimal solution for this equation is got by implementing $I_1 I_2 I_2^2 I_3^{-2} I_3^{-2} = 1$ or, equivalently, as $I_3 = I_1^{1/2} I_2^{1/2} I_2^1 I_3^{-1}$. Clearly, the presence of the $1/2$ means that 2 components are needed for the synthesis. Two slightly different networks result. The Coates graph $G_c(X)$ of the reduced input-connectivity matrices of these solutions along with the relevant powers is shown in (a) and (c). The corresponding complete MITE networks are shown in (b) and (d). The presence of the current mirrors makes the general case of rational powers different from the case where the powers are integral.

need to synthesize $I_1 I_2^3 I_3^{-4} = 1$. It is easy to see that the only minimal solution is obtained by splitting this as $I_1 I_2 I_2^2 I_3^{-2} I_3^{-2} = 1$ or as $I_3 = I_1^{1/2} I_2^{1/2} I_2^1 I_3^{-1}$. This we implement using the methods for synthesizing Λ matrices with $\pm 1/2$ powers and the resulting two possible solutions are shown in Figure 4.11.

4.7 2-MITE synthesis for partially reconfigurable POPL networks : The MITE FPAA

The Field-Programmable Analog Array (FPAA) is the analog counterpart of the FPGA. Multiple analog blocks can be connected in different ways to implement different analog functions. In the context of MITEs, FPAAs made of MITEs are discussed in [59, 60, 61].

An important question that arises with reconfigurable MITE circuits is the level of granularity to which the circuits need be made reconfigurable. In other words, we would like to have some fixed blocks that are non-reconfigurable and different functions are obtained by reconnected the input and output terminals of these fixed blocks with those of other fixed blocks. Making the fixed blocks smaller (in the limit, it is a MITE or a capacitor or a current mirror) usually increases the number of synthesizable functions but has the disadvantage of increasing the size of the "switch-matrix" needed to implement these functions. On the other hand, making the fixed blocks larger reduces the size of the switch-matrix at the cost of the number of synthesizable functions.

A POPL network is an essential block in synthesizing any MITE network. However, if we make a POPL network a fixed block in the MITE FPAA, the functional relationship between the inputs and outputs is fixed. We would, therefore, like to impart some minimal reconfigurability to the POPL network but also not take it to the limit of a single MITE. Instead of making the POPL network itself a fixed block, let us consider the option of making the *input* section of a POPL network, consisting of the input MITEs, fixed. In the case of a single-output POPL network which we will be concerned with in this section, different POPL functions can be obtained by connecting the gates of the output MITE differently into the input section.

The problem that is solved in this section is to find that optimal input 2-MITE network (i.e., the optimal input-connectivity matrix X) that gives us the maximum number of synthesizable functions. Mathematically speaking, for a given $X \in \mathcal{M}_n(\mathbb{N})$ that corresponds to the input connectivity matrix of a 2-MITE POPL network (i.e., $\det(X) \neq 0, \text{diag}(X) > 0, X\mathbf{1}_n = 2\mathbf{1}_n$), we consider $S = \{\Lambda = YX^{-1} \mid Y \in \mathcal{M}_{1\times n}(\mathbb{N});\ Y\mathbf{1}_n = 2\}$ which represents the set of synthesizable functions. We would like this to be maximal in some sense. Note that the cardinality of S is usually n^2 and is independent of X and so that cannot help us in distinguishing two synthesizable functions. However, it should be noted that in a MITE FPAA, we can always change the order of input currents as we want to. Hence, the ability to synthesize $I_1^2 I_2 I_3^{-2}$ and $I_1^2 I_2^{-2} I_3$ is no better than the ability to synthesize $I_1^2 I_2 I_3^{-2}$, since we would simply exchange I_2 and I_3 to obtain the latter function. Hence, we need to take the synthesizable functions modulo permutations. In other words, we need to distinguish between functions that have distinct $(p_2, p_{-2}, p_1, p_{-1})$ where p_2, p_{-2}, p_1 and p_{-1}

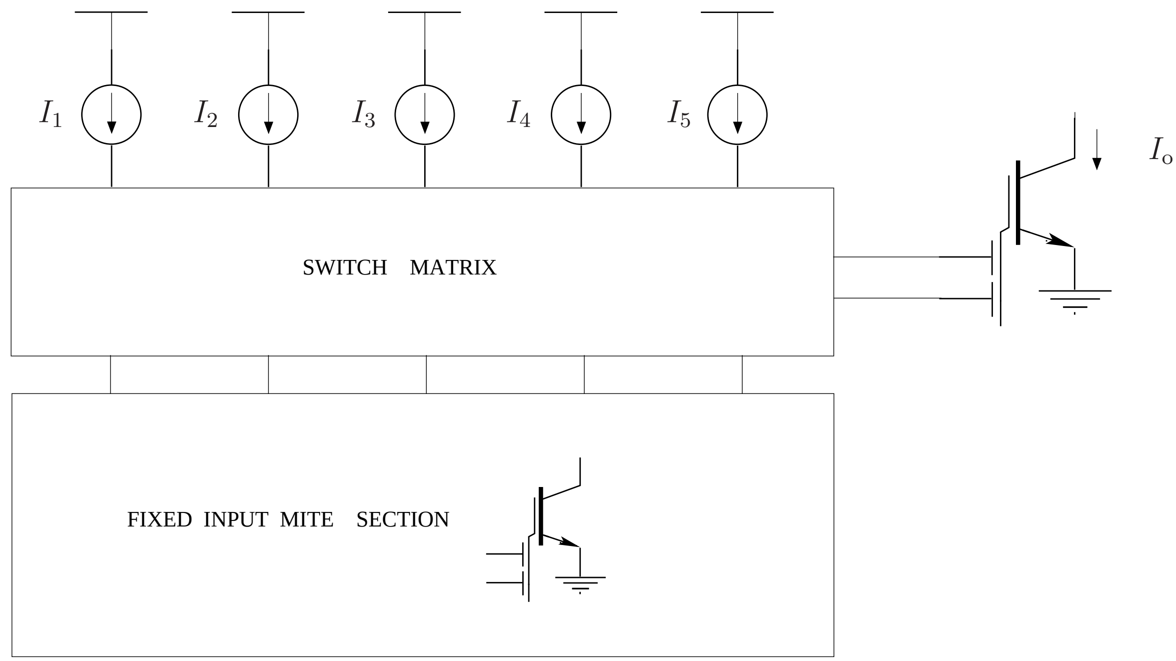


Figure 4.12. The scheme used to maximize the number of synthesizable functions while avoiding fine granularity in the MITE FPAA. The input section of the POPL network is fixed. Multiple functions are obtained by changing the connectivity of the output MITE. This reconfiguration is done through a switch matrix.

are the number of $+2, -2, +1$ and -1 powers in the POPL function.

We will first solve the problem for the lower order cases $n = 3$ and $n = 4$. The optimum X for higher n becomes apparent from these cases.

The only possible connected 2-MITE POPL networks for $n = 3$ and $n = 4$ are shown in Figure 4.13 and Figure 4.14. The ability of these input sections to synthesize different functions when the output gates are connected suitably is shown in Table 4.1. A $\surd$ (resp. $\times$) means that the network in that column can (resp. cannot) implement the function in

Table 4.1. The functional synthesizability of 2-MITE networks for 3 and 4 inputs

$n = 3$			
Function	(a)	(b)	(c)
$[1 \;\; 0 \;\; 0]$	$\surd$	$\surd$	$\surd$
$[1 \;\; -1 \;\; 1]$	$\surd$	$\surd$	$\surd$
$[2 \;\; -1 \;\; 0]$	$\surd$	$\surd$	$[\surd]$
$[2 \;\; -2 \;\; 1]$	$\surd$	$\times$	$\times$

$n = 4$					
Function	(a)	(b)	(c)	(d)	(e)
$[1 \;\; 0 \;\; 0 \;\; 0]$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
$[1 \;\; -1 \;\; 1 \;\; 0]$	$\surd$	$\surd$	$\surd$	$\surd$	$\surd$
$[2 \;\; -1 \;\; 0 \;\; 0]$	$\surd$	$\surd$	$\surd$	$\surd$	$[\surd]$
$[2 \;\; -1 \;\; 1 \;\; -1]$	$\surd$	$\times$	$\times$	$\times$	$\times$
$[1 \;\; -2 \;\; 1 \;\; 1]$	$\times$	$\surd$	$\times$	$\times$	$\times$
$[2 \;\; -2 \;\; 1 \;\; 0]$	$\surd$	$\surd$	$\surd$	$\times$	$\times$
$[2 \;\; -2 \;\; 2 \;\; -1]$	$\surd$	$\times$	$\times$	$\times$	$\times$

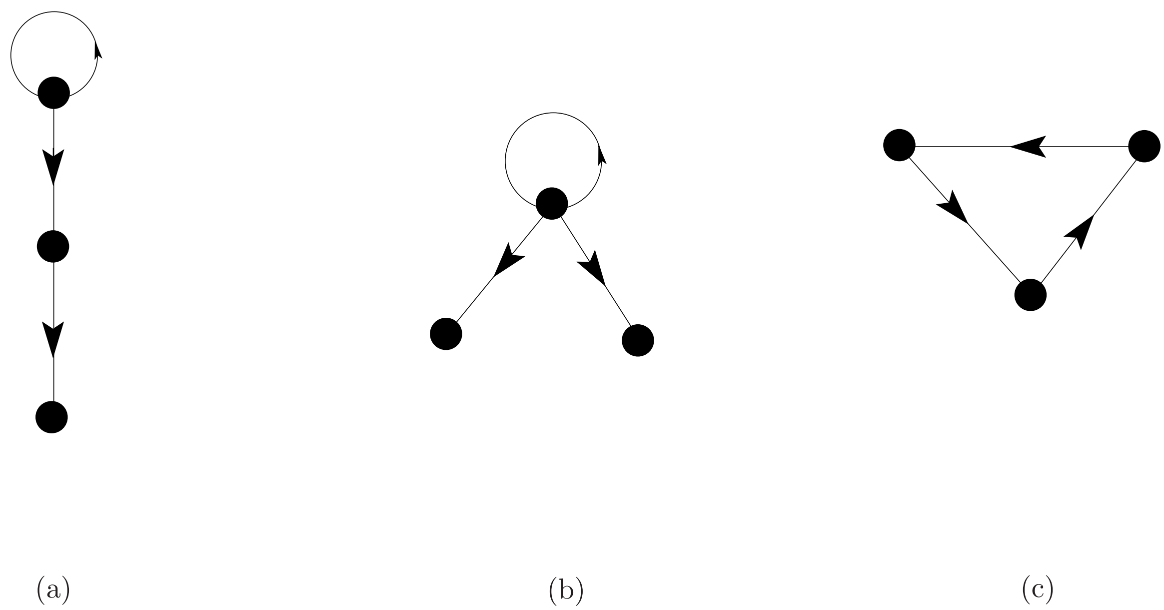


Figure 4.13. (a), (b), and (c) are the only connected 2-MITE POPL networks with 3 inputs i.e., $n = 3$. From Table 4.1, it follows that (a) can synthesize the maximum number of functions for $n = 3$, while (b) and (c) fall short : (b) and (c) cannot synthesize $I_o = I_1^2 I_2^{-2} I_3$ and (c) can implement $I_o = I_1^2 I_2^{-1}$ only by using copies of currents in $I_1 I_2 I_3^{-1}$.

that row. A [$\surd$] means that the function in that row can be implemented by the network in that column but only when copies of currents are allowed. For example, the function $\Lambda = [-1 \;\; 2 \;\; 0]$ can be implemented by (a) by choosing $Y = [0 \;\; 2 \;\; 0]$, by (b) by choosing $Y = [0\; 2\; 0]$, but can be implemented by (c) only by having $Y = [0 \;\; 2 \;\; 0]$ and by sending a copy of the current I_2 through 3 i.e., by having $I_3 = I_2$. From the above cases, it is intuitively clear that the best choice is an (a)-type structure. This is true in general as given by the following theorem:

Theorem 4.7.1 *The structure shown in Figure 4.15, which we call the basic structure, can implement any 2-MITEable single-output POPL function Λ with integer elements which has at most n inputs except those in which both the following conditions, which we collectively call C_n, are satisfied:*

1. *All the n inputs are raised to nonzero powers.*

2. *The sum of powers that are ± 2 is -2.*

C_n can be satisfied only if n is even. If C_n is not satisfied, then we say that Λ is implementable by the basic structure. *Further, a copy of a current is required for implementing*

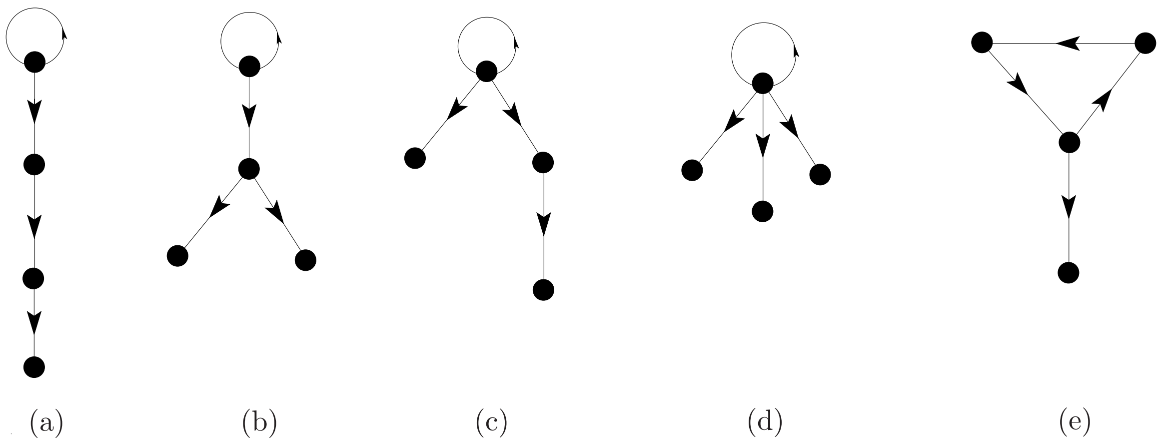


Figure 4.14. (a), (b), (c), (d), and (e) are the only connected 2-MITE POPL networks with 4 inputs i.e., $n = 4$. From Table 4.1, it follows that (a) can synthesize the maximum number of functions for $n = 4$, while everything else falls short : (b) and (c) cannot synthesize $I_o = I_1^2 I_2^{-2} I_3^2 I_4^{-1}$ and $I_o = I_1^2 I_2^{-1} I_3 I_4^{-1}$ and (c), in addition, cannot implement $I_o = I_1 I_2^{-2} I_3 I_4$ too. (d) and (e) have obviously very limited functional synthesizability.

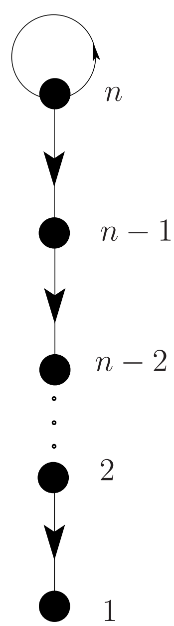


Figure 4.15. The basic structure used to implement almost any 2-MITEable single-output POPL function with at most n inputs.

the POPL function only when the sum of ± 2 *is* -2.

Proof: We give the proof by considering the following three cases:

Case 1: *The sum of powers in* Λ *that are* ± 2 *is* $+2$.

Since Λ is 2-MITEable, it follows that the sum of powers that are ± 1 is -1. It is clear that there is at least one power equal to $+2$ and one equal to -1. Let t be the number of powers that are "$+2$" and s the number of powers that are "-1". Then, there are $t-1$ "-2" powers and $s-1$ "$+1$" powers and the remaining $k = n - ((2t-1)+(2s-1))$ powers are 0 powers. That Λ is 2-MITEable follows from the following sequence which shows how the currents should be distributed from bottom to top in Figure 4.15:

$$\underbrace{0\ldots0}_{0-\text{powers}}\ \underbrace{+1\ -1\ldots-1}_{\pm1-\text{powers}}\ \underbrace{+2\ -2\ldots+2}_{\pm2-\text{powers}}\ -1 \tag{4.32}$$

Here, the output connectivity vector $\mathbf{y} = [y_j]$ is given by $y_j = \delta_{j,k+1} + \delta_{j,k+2s-1}$ i.e y_j is zero except at those points where the powers change from a "0" or "± 1" sequence to a "± 1" or "± 2" sequence, respectively.

Case 2: *The sum of powers in* Λ *that are* ± 2 *is* 0.

Since Λ is 2-MITEable, it follows that the sum of powers that are ± 1 is $+1$. It is clear that there is at least one power equal to $+1$. Let t be the number of powers that are $+2$ and s the number of powers that are 1. Then, there are t "-2" powers and $s-1$ "-1" powers and the remaining $k = n - ((2t)+(2s-1))$ powers are 0 powers. That Λ is 2-MITEable follows from the sequence:

$$\underbrace{0\ldots0}_{0-\text{powers}}\ \underbrace{+1\ -1\ldots-1}_{\pm1-\text{powers}}\ \underbrace{+2\ -2\ldots-2}_{\pm2-\text{powers}}\ +1 \tag{4.33}$$

The output connectivity vector $\mathbf{y} = [y_j]$ is given by $y_j = \delta_{j,k+1} + \delta_{j,k+2s-1}$ i.e., y_j is zero except at those points where the powers change from a "0" or "± 1" sequence to a "± 1" or "± 2" sequence, respectively.

Case 3: *The sum of powers in* Λ *that are* ± 2 *is* -2.

Since Λ is 2-MITEable, it follows that the sum of powers that are ± 1 is $+3$. Let t be the number of powers that are "-2" and "s" the number of powers that are $+1$. Then, there are $t-1$ "+2" powers and $s-3$ "−1" powers and the remaining $k = n-((2t-1)+(2s-3))$ powers are 0 powers. Clearly, the number of nonzero powers, $2(t+s)-4$ is even. Comparison with the previous two cases shows that the presence of three +1 powers means that the powers cannot be arranged into the basic structure as they are given. The obvious way to tackle this is to convert one of the −2 powers into −1 powers by splitting I^{-2} as $I^{-1}I^{-1}$. This reduces the problem to the previous case; however, the number of nonzero powers is also increased by 1. Therefore, if C_n is satisfied, we will not be able to implement Λ by the basic structure but in case the number of nonzero powers is lesser than n (i.e., $k > 0$), this can be accommodated into the previous case. Hence, the theorem follows. □

4.8 Coates graph analysis of general POPL networks

The graph-theoretic analysis of POPL networks that is being presented in this section is not restricted to 2-MITE networks alone. The purpose of the presentation is to smooth the way to the discussion about 2-output 2-MITE networks to be presented in the next section.

As shown in Chapter 3, any power matrix $\Lambda \in \mathcal{M}_{p,n}(\mathbb{Q})$ can be implemented using a POPL network described by two connectivity matrices $X \in \mathcal{M}_n(\mathbb{N})$ and $Y \in \mathcal{M}_{p,n}(\mathbb{N})$. We have the relation $Y = \Lambda X$. Writing Λ and Y in terms of their columns, $\Lambda = [\Lambda_1\ \Lambda_2 \ldots \Lambda_n]$ and $Y = [\mathbf{y}_1\ \mathbf{y}_2 \ldots \mathbf{y}_n]$, we have $\mathbf{y}_j = \sum_{k=1}^{n} \Lambda_k x_{kj}$. Let us consider the Coates graph $G_{\mathrm{c}}(X)$ of X. Each vertex k in $G_{\mathrm{c}}(X)$ can be associated with Λ_k and a $\mathbf{y}_k$. It can be seen that

$$\Lambda_j = (\mathbf{y}_j - \sum_{k \neq j}^{n} \Lambda_k x_{kj})/x_{jj} \tag{4.34}$$

The Λ_j associated with a vertex j, therefore, depends upon the Λ_k associated with its "successors" i.e., the vertices with nonzero x_{kj}, and also an "input" $\mathbf{y}_j$. We will refer to Λ_j as the *power* of j. If $\mathbf{y}_j \neq 0$, we will say that there is a *source* of value $\mathbf{y}_j$ at j.

Let us now apply this analysis to a 2-MITE POPL network. The "usual" value of x_{jj} is 1; the case $x_{jj} = 2$ can be viewed as a degenerate case of the directed circuit becoming a self-loop. Any "successor" clearly has $x_{kj} = 1$. The successors of j form the set $\alpha^{-1}(j) = \{k \mid \alpha(k) = j\}$, where $\alpha(k)$, by definition, is the unique parent of k. Hence,

$$\Lambda_j = \mathbf{y}_j - \sum_{k \in \alpha^{-1}(j)} \Lambda_k \tag{4.35}$$

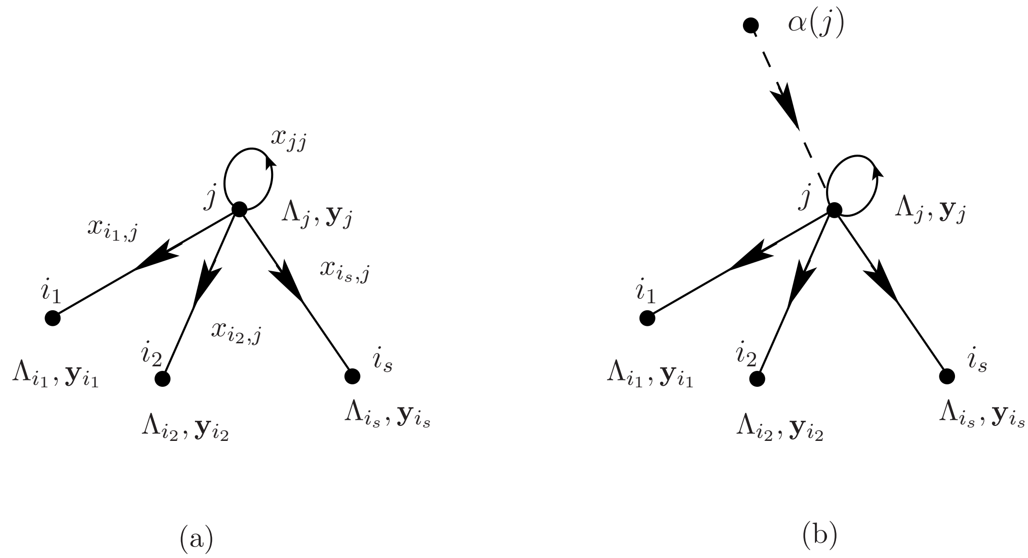


Figure 4.16. (a) The vertex j and its successors $i_1, i_2, \ldots, i_s$ in the Coates graph of a general POPL MITE network. The power of j, Λ_j, is related to the powers of $i_1, i_2, \ldots i_s$ and to the source at j through Equation (4.35). (b) The graph of (a) for the particular case when the fan–in is 2. In this case, every vertex j has a unique parent $\alpha(j)$. Barring the degenerate case of C being a loop, the weights of all edges is now unity .

The above equation gives us a simple, intuitive method to find the Λ_j at every vertex j if the graph of a 2-MITE POPL network is given and the source values y_j are known. The procedure is to find the vertices of zero out-degree and starting from these vertices to proceed to the other vertices by successively applying Equation (4.35), noting the fact that for vertices with zero out-degree, $\Lambda_j = \mathbf{y}_j$. It should be noted that sources are generally absent in many vertices if the number of input MITEs is much larger than the number of output MITEs. The general case as well as the 2-MITE cases are shown in Figure 4.16.

While the Coates graph itself is that of the reduced input connectivity matrix $\widehat{X}$, we can make it represent the whole POPL network by associating the pair $(\Lambda_j, \mathbf{y}_j)$ with each vertex j. From Equation (4.35), it is enough to give y_j at each vertex j, we give Λ_j for clarity in the presentation below. We call this also the Coates graph of the POPL network; the distinction being usually clear.

Suppose we do not know the power of an immediate descendant k' of j but only that of k''s descendants. Assuming that there is no source at k', Equation (4.35) changes to

$$\Lambda_j = \mathbf{y}_j + (-1) \sum_{\substack{k \in \alpha^{-1}(j) \\ k \neq k'}} \Lambda_k + (-1)^2 \sum_{k'' \in \alpha^{-1}(k')} \Lambda_{k''} \tag{4.36}$$

Proceeding this way, it can be shown inductively that if $i_1, i_2, \ldots, i_t$ are descendants of j such that

1. No two vertices amongst $i_1, i_2, \ldots, i_t$ have a descendent-ancestor relationship

2. Any other descendent of j is either an ancestor or a descendent of at least one of $i_1, i_2, \ldots, i_t$,

then

$$\Lambda_j = \sum_{k=1}^{t} (-1)^{d(i_k,j)} \Lambda_{i_k} + \sum_{s} (-1)^{d(s,j)} \mathbf{y}_s \tag{4.37}$$

where $d(i_k, j)$ is the number edges in the path from j to i_k and the second sum is over all the vertices s with a nonzero source in the paths from j to the i_k's.

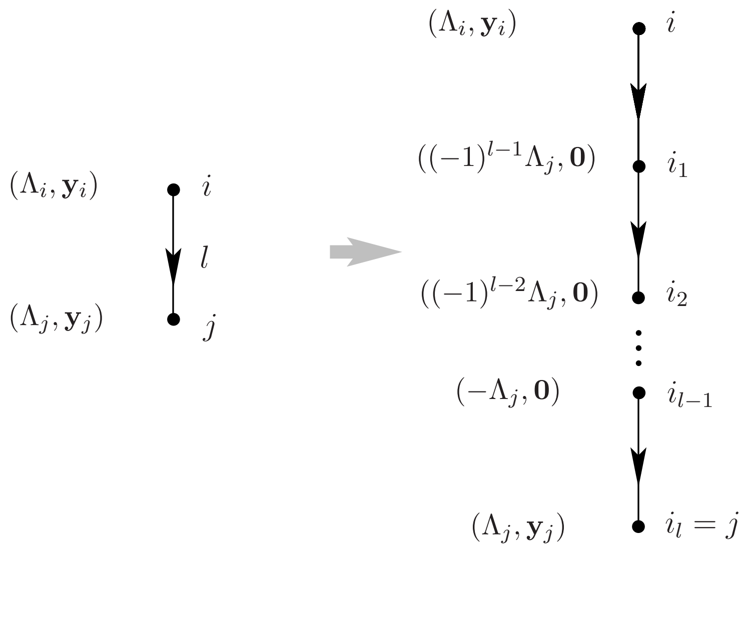
This formalism helps us in another way in analyzing 2-MITE networks:

If all the vertices in the interior of the path from j to i of length l have no sources and have in-degree equal to out-degree, then we can replace the whole path with a single edge and associate a property "length" l with the edge, always noting that the intermediate vertices have powers that is equal to Λ_i with alternating signs. Hence, the *modified Coates graph* representation (MCGR) of a POPL MITE network can be arrived at by replacing all such paths with edges in the Coates graph of X. Formally,

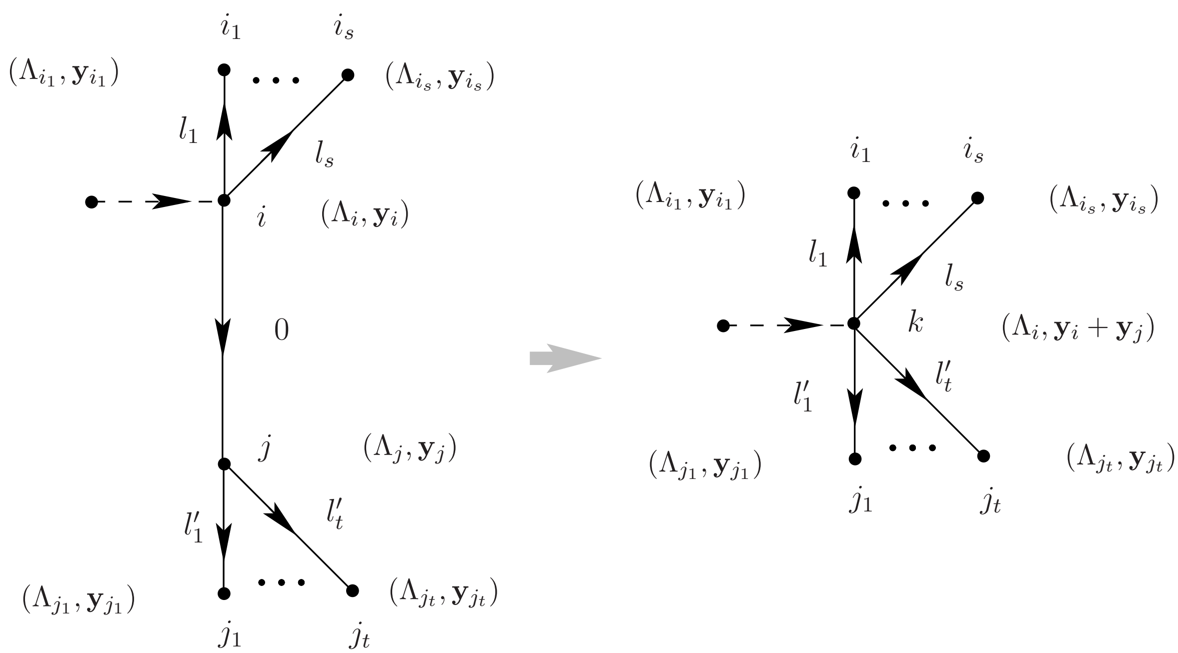
Definition 4.8.1 *A modified Coates graph representation (MCGR) of a 2-MITE POPL network is a digraph G^* satisfying the conditions of Theorem 4.3.1 with "lengths" $l(e) \geq 0$ associated with each edge e and a pair $(\Lambda_j, \mathbf{y}_j) \in \mathbb{Q}^p \times \mathbb{N}^p$ associated with each vertex j so that it transforms into the Coates graph of the POPL network when the following replacements are made for each edge $e = (i, j)$:*

1. *If $l = l(e) > 0$, then e is replaced by a directed path $i, i_1, i_2, \ldots, i_l = j$ with $\mathbf{y}_{i_s} = \mathbf{0}$ and $\Lambda_{i_s} = (-1)^{l-s}\Lambda_j$ when $s \in [1 : l-1]$. This is shown in Figure 4.17(a).*

2. *If $l(e) = 0$, then the edge e is* contracted, *i.e., i and j are replaced by a new vertex k whose neighbors are those of i and j with $\Lambda_k = \Lambda_i$ and $y_k = y_i + y_j$. This is shown in Figure 4.17(b).*



(a)



(b)

Figure 4.17. Transformation of a MCGR of a POPL network into the Coates graph of the network.

Note that we could have defined the length $l(e)$ to be the property of the tail vertex of e itself, since each vertex has in-degree 1. Therefore we will sometimes write $l(j)$ to mean $l\left((\alpha(j), j)\right)$. It is clear that with this definition, Equation (4.35) now transforms to

$$\Lambda_j = \mathbf{y}_j + \sum_{k \in \alpha^{-1}(j)} (-1)^{l(k)} \Lambda_k \tag{4.38}$$

where all the properties are with respect to a MCGR of the POPL network. It is also easily seen that a MCGR of a POPL network need not be the *only* MCGR of that network. However, we now prove the following:

Theorem 4.8.1 *There is a MCGR G^* of any 2-MITE POPL network such that in G^**

1. *All vertices have out-degree either* 0 *or* 2.
2. *If a vertex j has out-degree* 2*, then it has no source, i.e.,* $\mathbf{y}_j = 0$.
3. *The source $\mathbf{y}_j$ of any vertex j with out-degree* 0 *is either the zero vector or an unit vector i.e., it has at most one nonzero element which is unity.*

Proof: First, if all the vertices in the interior of the path from j to i of length l have no sources and have in-degree equal to out-degree, then the whole path is replaced by a single edge of length l. This ensures that there are no vertices of out-degree 1 that have zero source.

Now, the transformations shown in Figure 4.18 are performed in sequence from top to bottom. In other words, repeated application of the first transformation results in a MCGR in which all vertices have out-degree at most 2. It should be noted that the first transformation results in a out-degree 2 with zero source. Now, if the resultant MCGR has any out-degree 2 vertex with nonzero source, it is transformed into vertices that satisfy this condition using the second transformation. The vertices shown with a circle around them denote those vertices that have out-degree 0. The only vertices that are now left are those with out-degree 1 or 0. All vertices with out-degree 1 and zero source have already been eliminated. If a vertex has nonzero source and out-degree 1, then it is transformed according to the third step shown in Figure 4.18 into a out-degree 2 vertex with zero source. It should be noted that in this sequence, the MCGRs do not have any vertices requiring an earlier step. Finally, if a vertex i has zero out-degree and has a source that is not the zero vector or a unit vector, then it is transformed according to the last step. Here, k is a index for which $\mathbf{y}$ is nonzero i.e., $\mathbf{y}(k) \neq 0$, from which it follows that $\mathbf{y}' = \mathbf{y} - \mathbf{e}_k$ is a positive vector that has entries in $\{0, 1, 2\}$. These steps show the construction of a MCGR of a POPL network satisfying Theorem 4.8.1. $\square$

What Theorem 4.8.1 essentially states is that in order to find the MCGRs corresponding to 2-MITE POPL networks with say, k, outputs, we need to look for those Coates graphs satisfying Theorem 4.3.1 which have out-degree 2 at all vertices except $2k$ vertices at which the out-degree is zero. To give an example, it is easy to see that there are essentially only three distinct forms of MCGRs for single-output POPL networks, as shown in Figure 4.19. The different values of l_1, l_2, l_3 and l_4 determine the association of vertices with different powers. The only thing to be noted that l_1 should be odd in Figure 4.19(a) and $l_1 + l_2$ must be odd in Figure 4.19(b). Similarly, both l_1 and l_3 must be odd in Figure 4.19(c).

4.9 2-MITE POPL networks with two outputs

In this section, an attempt is made to characterize POPL networks made of 2-MITEs with two outputs. Let $\Lambda \in \mathcal{M}_{2,n}(\mathbb{Q})$ be a given power matrix. It is clear that a necessary condition for Λ to be 2-MITEable is that each row of Λ should be 2-MITEable separately, which is easily checked. Using Theorem 4.8.1, we first create a catalog of the only possible MCGRs of 2-MITE POPL networks with 2-outputs. The results are presented in Figure 4.20, where the MCGRs are connected and in Figure 4.21, where the MCGRs are not connected digraphs. For simplicity's sake, we will restrict the discussion to the connected graphs in Figure 4.20 as they produce powers that are only $\pm 2, \pm 1$ or 0. Even the MCGR in Figure 4.21(a), 4.21(b), and 4.21(c) can produce nonfractional powers; however, in that case the power matrix Λ can be separated out as $\left(\begin{smallmatrix} \Lambda_1 & 0 \\ 0 & \Lambda_2 \end{smallmatrix}\right)$ and it is enough to check for the 2-MITEability of each row of Λ for the whole matrix to be 2-MITEable. The structures in Figure 4.20 and Figure 4.21 are the only possible graphs that a MCGR of any 2-MITE 2-output POPL network can take. However, to convert them into actual MCGRs, the lengths of the edges as well as the sources at the vertices of zero out-degree needs to be specified. It is clear that this leads to numerous different power matrices. A catalog

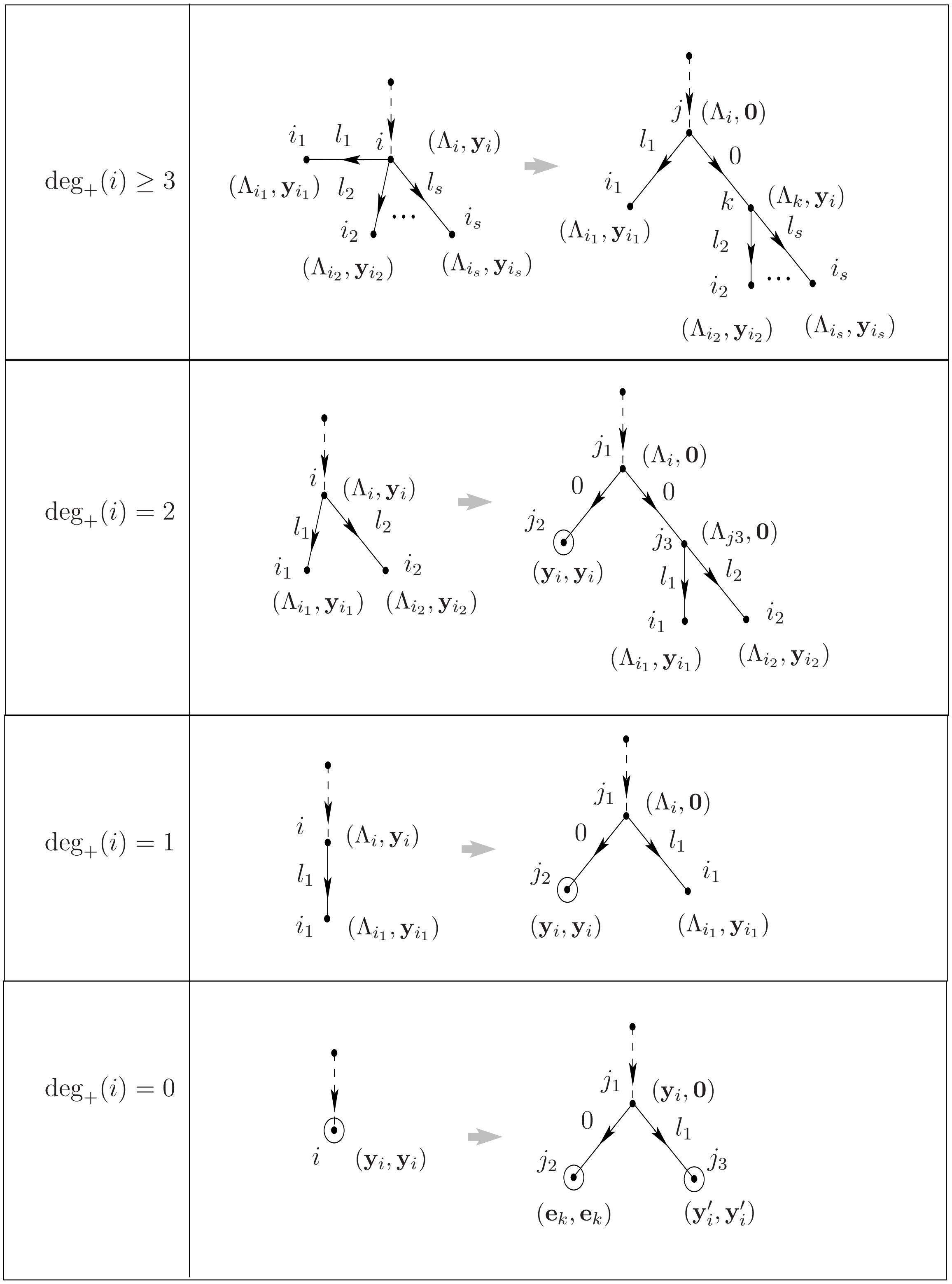


Figure 4.18. Construction of a MCGR of a 2-MITE POPL network satisfying Theorem 4.8.1.

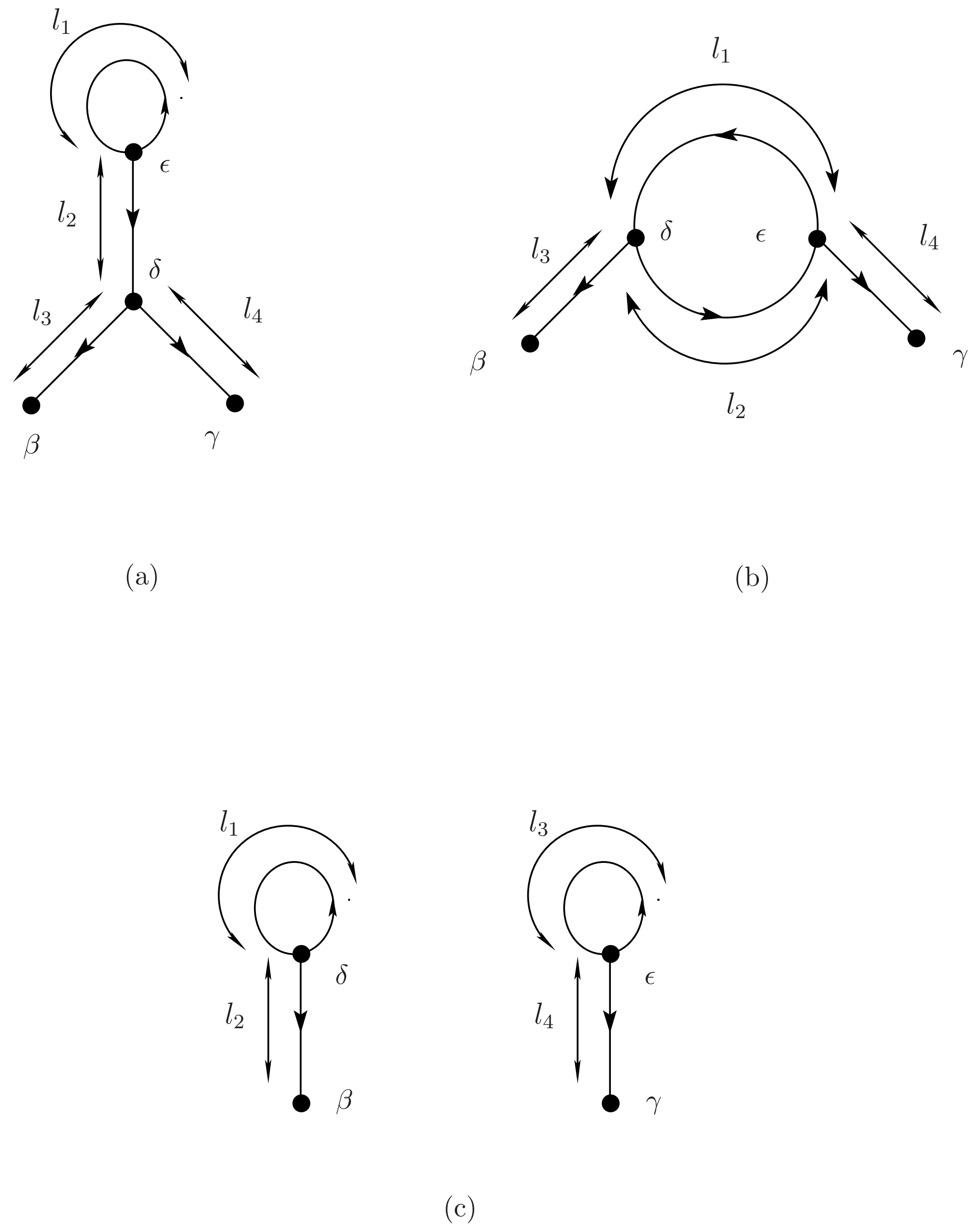


Figure 4.19. The modified Coates graphs of the only structurally distinct Coates graphs of X for the single-output case. The length of each edge is also mentioned. Note that the effect of the powers of the successors of j can be calculated from Equation (4.37)

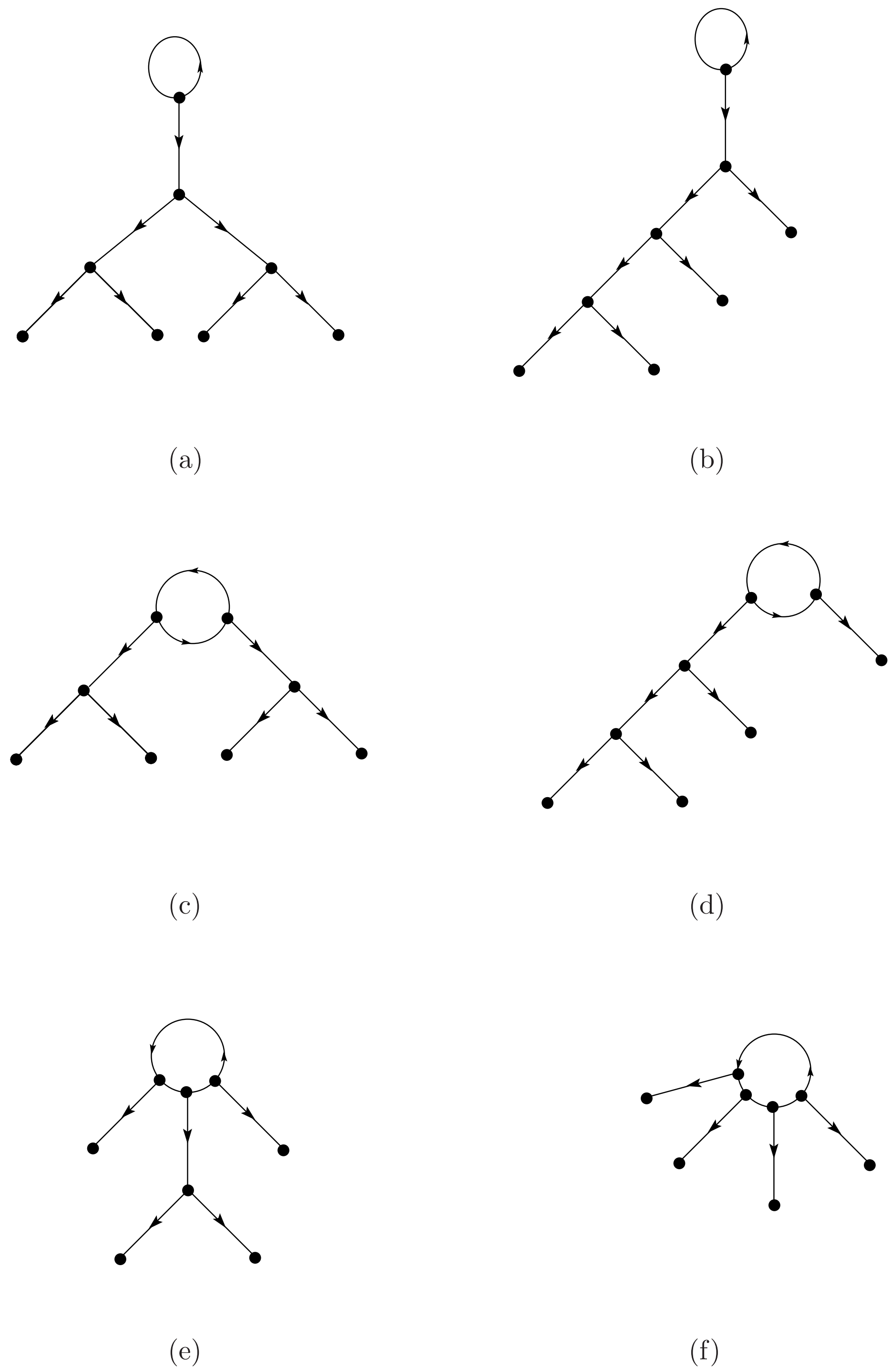


Figure 4.20. The only possible connected distinct graphs representing the MCGRs of 2-MITE POPL networks with two outputs. The elements of the power matrices are all in $\{0, \pm 2, \pm 1\}$: no $\pm 1/2$ powers exist.

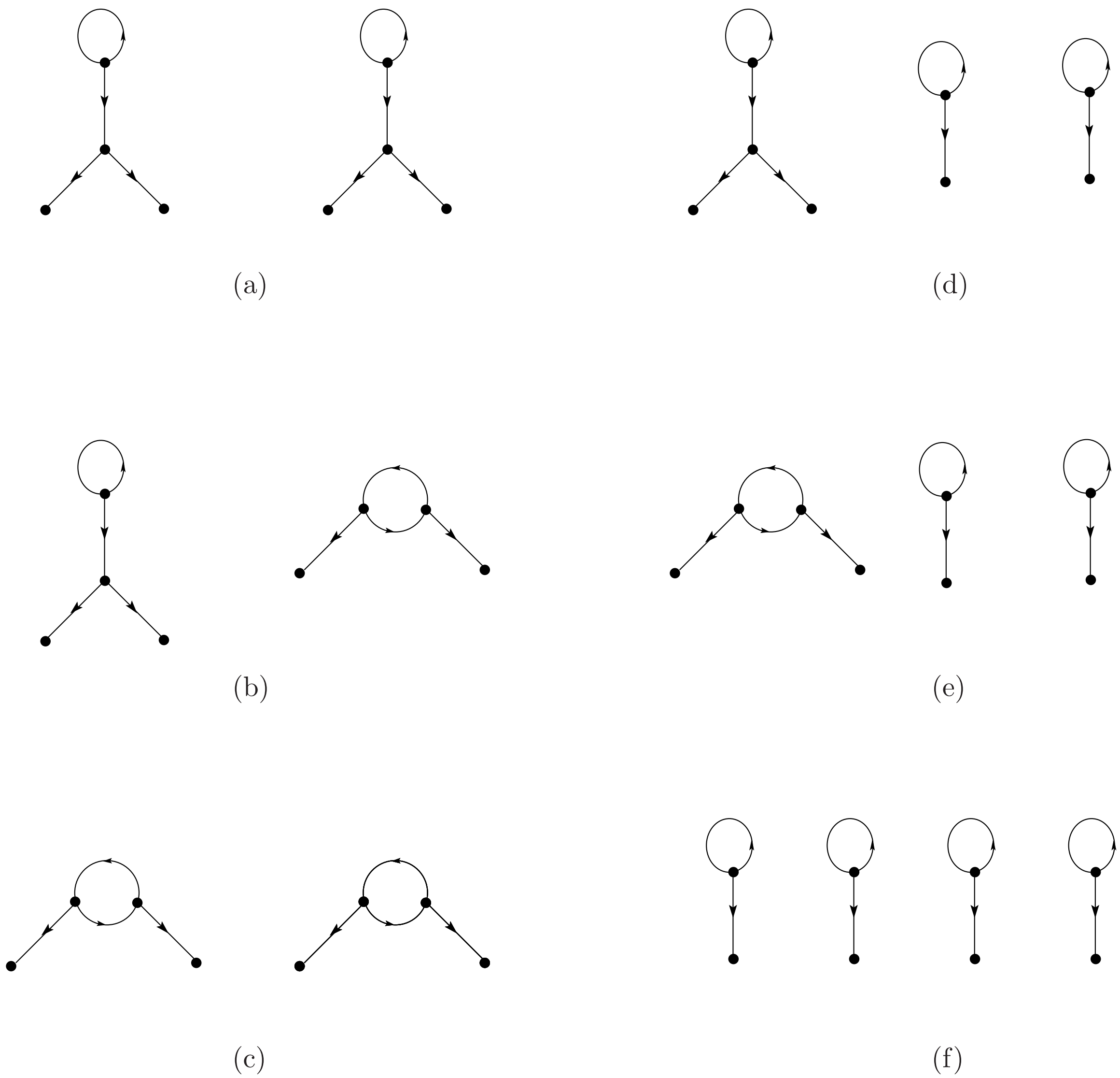


Figure 4.21. The only possible disconnected MCGRs of 2-MITE POPL networks with two outputs. The power matrices necessarily have fractional ($\pm 1/2$) powers, except in (a),(b), and (c) under some conditions.

of the different powers that can result from the graphs in Figure 4.20(a) and (c) are given in Figure 4.22. The powers generated by Figure 4.20(b) and (d) are given respectively at Figure 4.23 and Figure 4.24, respectively. These powers are derived from the basic graphs by assuming different lengths of the edges in the graphs. Some necessary conditions for Λ to be 2-MITEable which follow from these graphs are as follows:

1. Λ cannot contain both $\left[\begin{smallmatrix}\pm 2\\ \pm 1\end{smallmatrix}\right]$ and $\left[\begin{smallmatrix}\pm 1\\ \pm 2\end{smallmatrix}\right]$. Here, $\left[\begin{smallmatrix}\pm 1\\ \pm 2\end{smallmatrix}\right]$ refers to any one of $\left[\begin{smallmatrix}1\\ 2\end{smallmatrix}\right]$, $\left[\begin{smallmatrix}1\\ -2\end{smallmatrix}\right]$, $\left[\begin{smallmatrix}-1\\ 2\end{smallmatrix}\right]$, and $\left[\begin{smallmatrix}-1\\ -2\end{smallmatrix}\right]$.
2. An element of $\{\pm\left[\begin{smallmatrix}2\\ 1\end{smallmatrix}\right], \pm\left[\begin{smallmatrix}1\\ 2\end{smallmatrix}\right], \pm\left[\begin{smallmatrix}2\\ 2\end{smallmatrix}\right]\}$ as a column in Λ cannot exist as a column of Λ if an element of $\{\pm\left[\begin{smallmatrix}2\\ -1\end{smallmatrix}\right], \pm\left[\begin{smallmatrix}1\\ -2\end{smallmatrix}\right], \pm\left[\begin{smallmatrix}2\\ -2\end{smallmatrix}\right]\}$ is a column of Λ and vice versa.
3. $\pm\left[\begin{smallmatrix}2\\ 2\end{smallmatrix}\right]$ and $\pm\left[\begin{smallmatrix}1\\ -1\end{smallmatrix}\right]$ cannot both be columns of Λ. Similarly, $\pm\left[\begin{smallmatrix}2\\ -2\end{smallmatrix}\right]$ and $\pm\left[\begin{smallmatrix}1\\ 1\end{smallmatrix}\right]$ cannot both be columns of Λ.

4.10 Appendix 4.A

In this appendix, we prove the following theorem:

Theorem 4.10.1 *The input connectivity matrix X of a 2-MITE POPL network satisfying Assumption 1 is diagonally stable.*

To recall the definition given in Chapter 2,

Definition 4.10.1 *A matrix $M \in \mathcal{M}_n(\mathbb{R})$ is said to be diagonally stable if it has a positive diagonal Lyapunov solution i.e., there exists a diagonal matrix $P > 0$ such that $PM + M^T P$ is positive definite.*

Proof of Theorem 4.10.1: The proof will be given in four steps:

Step 1 We will show that the input connectivity matrix X can be transformed by a simultaneous permutation of rows and columns into the form

$$\begin{bmatrix} X_1 & 0 \\ X_2 & X_3 \end{bmatrix} \tag{4.39}$$

where X_1 is a circulant matrix of a particular form (for the definition of circulant matrices, see [28]) and X_3 is a *acyclic* matrix, i.e., a square matrix whose associated

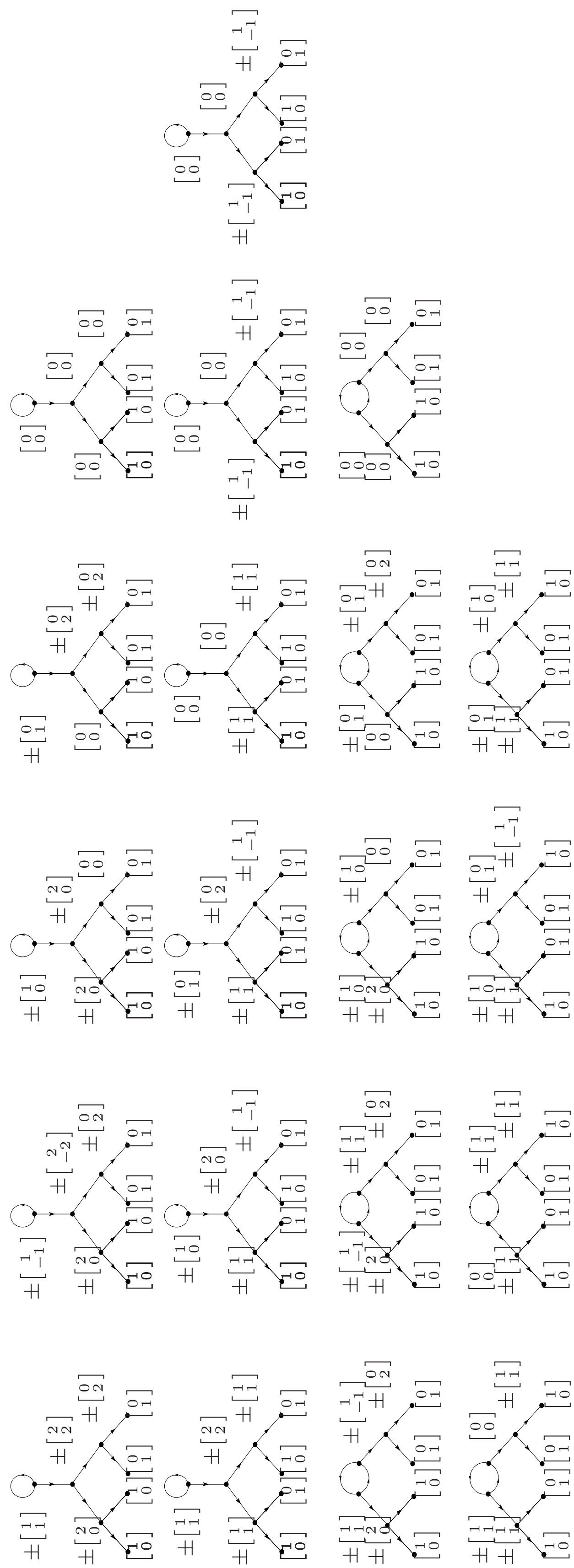


Figure 4.22. Different powers generated by the MCGRs in Figure 4.20(a) and (c).

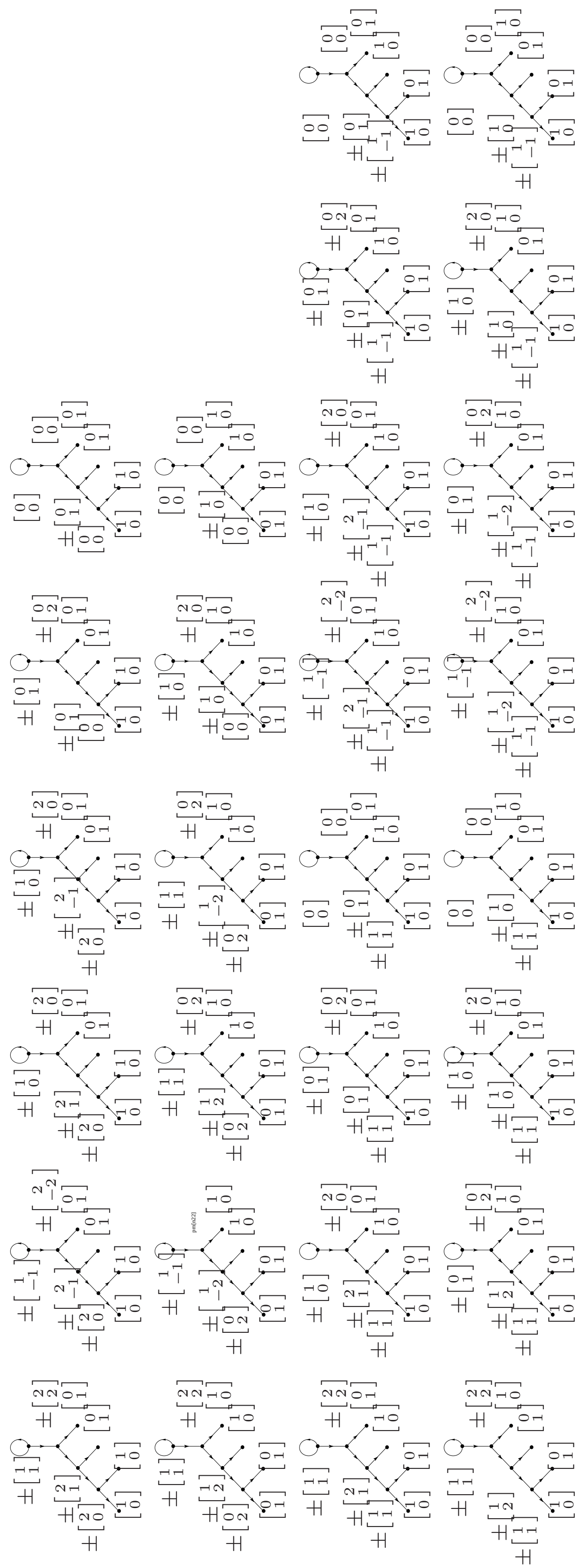


Figure 4.23. Different powers generated by the MCGR in Figure 4.20(b).

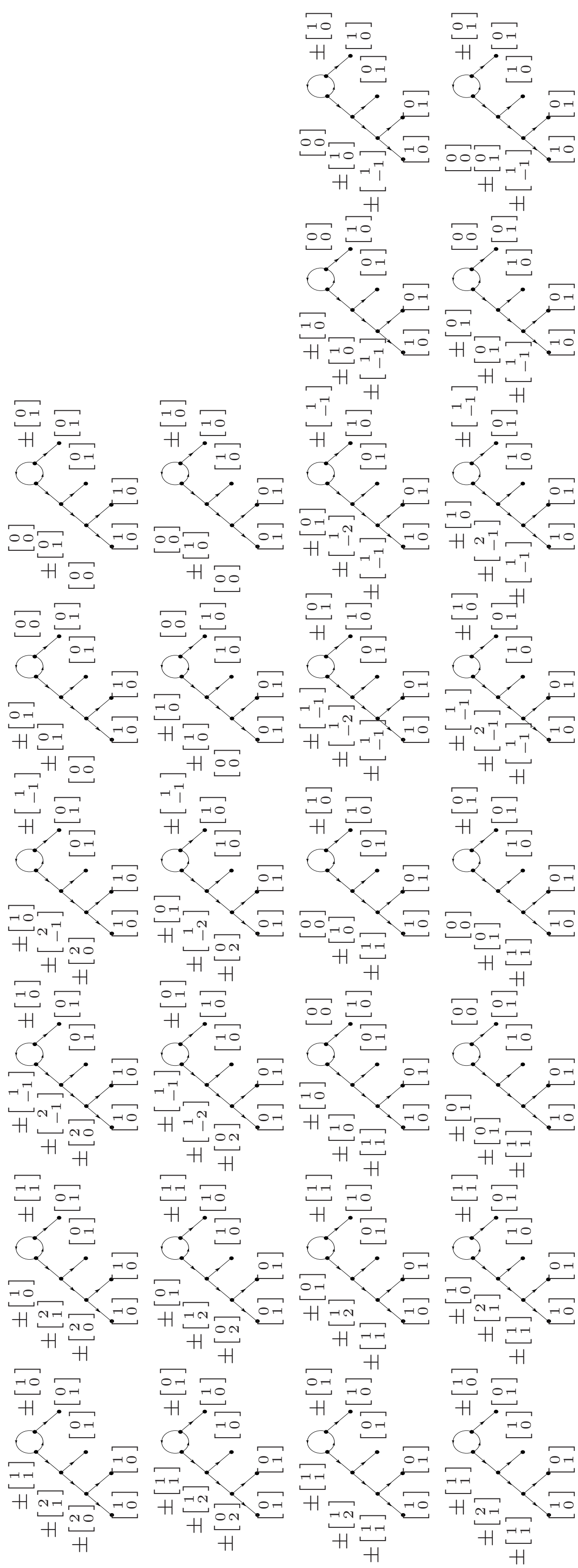


Figure 4.24. Different powers generated by the MCGR in Figure 4.20(d).

Coates graph is such that the underlying undirected graph is a forest, barring self-loops. Without loss of generality, we will assume that X itself is of the form in Equation (4.39).

Step 2 X is diagonally stable if and only if X_1 and X_3 are diagonally stable. This fact has been proved in [62].

Step 3 We will show that X_1 is diagonally stable. Here the assumption that the directed circuit C is of odd length is crucial.

Step 4 We will show that X_3 is a P-matrix. An acyclic P-matrix is diagonally stable [62].

Step 1: First, it must be noted that if X is a direct sum of matrices that are diagonally stable, then X is also diagonally stable. If Q is a permutation matrix, then X is diagonally stable, D-stable, or is a P_0 matrix if and only if QXQ' also has the same property. This means that we can reorder the rows and the columns similarly without affecting any property of the MITE network. Hence, we can write X as a direct sum of matrices that are connected, each representing the components of $G_c(X)$. Thus, without loss of generality we can assume that $G_c(X)$ is connected and, by Theorem 4.3.1, it follows that there is a unique directed circuit C associated with $G_c(\widehat{X})$.

Vertex ordering convention: For purposes of this proof, we will follow the following convention:

1. The vertices in the directed circuit C of length k are numbered $1, 2, \ldots, k$, with 2 being the ancestor of 1, and so on.

2. Every other vertex k satisfies $k > \alpha(k)$ i.e., all the ancestors of k are indexed with a number lower than k in the usual ordering of integers.

By Theorem 4.3.1, if the directed circuit is contracted to a vertex v, the resultant graph is a rooted tree with v as the root. Since there are no edges directed *to* v *from* the remaining vertices in the tree, it is clear that in $X = [x_{ij}]$, $x_{ij} = 0$ if $i \in [1 : l]$ and $j \notin [1 : l]$. Hence it is clear that X is of the form in Equation (4.39) with X_1 being a square matrix of order k and X_3 a square matrix of order $n - k$.

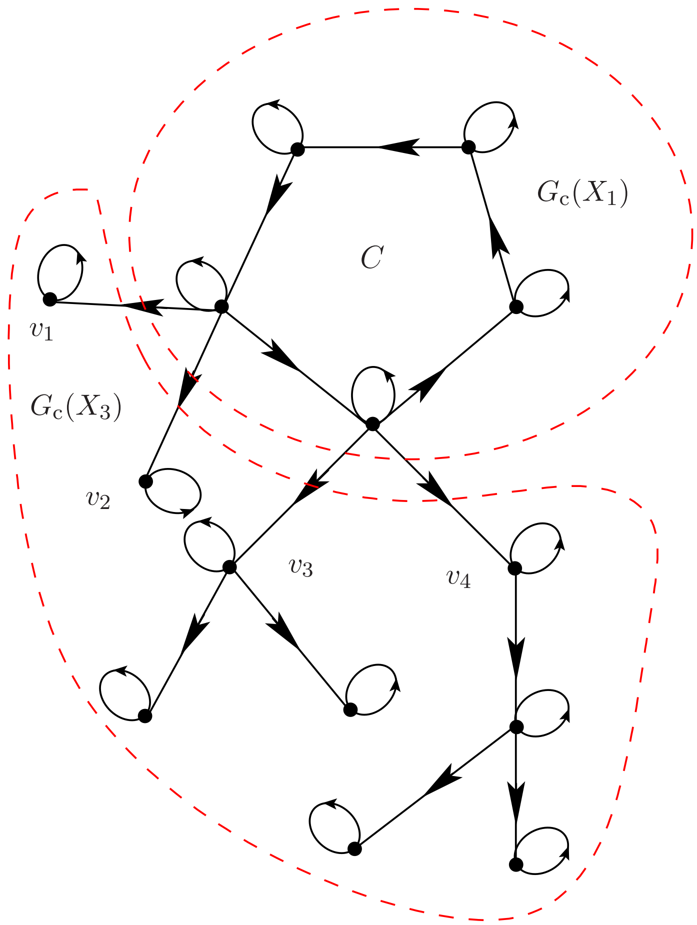


Figure 4.25. The input-connectivity matrix X can be written, by a permutation similarity, in the form shown in Equation (4.39). $G_c(X_1)$ represents the directed circuit C, $G_c(X_3)$ represents $G_c(X) - C$, $G_c(X_2)$ represents the edges connecting $G_c(X_1)$ and $G_c(X_3)$.

By the vertex ordering chosen, it is clear that for $i = 1, 2, \ldots, k-1$, the only nonzero elements in the i^{th} row are x_{ii} and $x_{i,i+1}$ and are all 1). Further, $x_{kk} = x_{k1} = 1$. Hence, we have

$$X_1 = \begin{bmatrix} 1 & 1 & 0 & \cdots & 0 \\ 0 & 1 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 & 1 \\ 1 & 0 & \cdots & 0 & 1 \end{bmatrix} \tag{4.40}$$

It should be noted that X_3 represents the edges connecting the vertices in C with those not in C as shown in Figure 4.25. Consider X_2. X_2 is obtained from $G_c(X)$ by deleting C and the edges connecting C and the remaining vertices i.e., X_2 corresponds to $G_c(X) - C$.

It is clear that deleting C creates as many rooted trees as there are edges between C and the remaining vertices, with roots $v_1, v_2, \ldots$ as shown in Figure 4.25. Clearly, $G_{\mathrm{c}}(X_2)$ does not have any circuits, barring the self-loops, since the only circuit in $G_{\mathrm{c}}(\widehat{X})$ is C. Hence X_2 is an acyclic matrix.

Step 2: The fact that the diagonal stability of a block triangular matrix depends only on its diagonal blocks is mentioned in [63] and is proved in [62]. The proof of this theorem requires a result from [64] that gives the following characterization of diagonally stable matrices: A matrix A is diagonally stable if and only if BA has a positive diagonal element for every nonzero positive semidefinite matrix B.

Step 3: We will show that $X_1 + X_1^T$ is positive definite, which clearly makes X_1 diagonally stable. For this, we need to show that $\mathbf{u}^T X_1 \mathbf{u} = \mathbf{u}^T (X_1 + X_1^T)\mathbf{u} > 0$ for all nonzero $\mathbf{u} = [u_i] \in \mathcal{M}_k$.

$$\begin{aligned}
\mathbf{u}^T X_1 \mathbf{u} &= \sum_{i=1}^{k} u_i \sum_{j=1}^{k} [X_1]_{ij} u_j \\
&= \sum_{i=1}^{k-1} u_i(u_i + u_{i+1}) + u_k(u_k + u_1) \\
&= \sum_{i=1}^{k} u_i^2 + \sum_{i=1}^{k-1} u_i u_{i+1} + u_k u_1 \\
&= \frac{1}{2}\left\{\sum_{i=1}^{k-1}(u_i^2 + u_{i+1}^2 + 2u_i u_{i+1}) + u_k^2 + u_1^2 + 2u_k u_1\right\} \\
&= \frac{1}{2}\left\{\sum_{i=1}^{k-1}(u_i + u_{i+1})^2 + (u_1 + u_k)^2\right\} \\
&\geq 0 \text{ for all } u \neq 0
\end{aligned} \tag{4.41}$$

$\mathbf{u}^T X_1 \mathbf{u} = 0$ if and only if for all $i \in [1 : k-1]$, $u_{i+1} = -u_i$ as well as $u_k = -u_1$. The first condition gives $u_i = (-1)^{i+1} u_1$ for all $i \in [1 : k]$, which implies $u_k = (-1)^{k+1} u_1$. Since the length of the directed circuit C, k, is odd by Theorem 4.3.1, we have $u_1 = u_k = -u_1$, which implies $u_1 = 0$, from which it follows that $\mathbf{u} = 0$. $\square$

Step 4:

Claim 4.10.1 X_3 *is a P-matrix*

Proof : Let $\widehat{X_3} = X_3 - I_{n-k}$. $G_c(\widehat{X_3})$ is a forest and since a direct sum of P–matrices is a P-matrix, it suffices to show that X_3 is a P-matrix when $G_c(X_3)$ is connected. By Theorem 4.3.1, the undirected graph underlying $G_c(\widehat{X_3})$ is a rooted tree, and hence three cases arise:

1. $n - k = 0$ i.e., X_3 is empty - Here there is nothing to prove since $X = X_1$.

2. $n - k = 1$ i.e., X_3 is a 1×1 matrix - Here it is clear that X_3 is a P-matrix since $X_3 = [1]$

3. $n - k \geq 2$.

Let $s = n - k$. We will prove the claim by induction on s. The base cases $s = 0, 1$ have been taken care of above. Let $s' \geq 2$ and let the claim be true for all nonnegative integers $s < s'$. We first claim that there is at least one vertex of out-degree 0 in $G_c(\widehat{X_3})$. To show this, let v be the root and consider the directed path P of largest length beginning from v and ending in the vertex (say) w. If w has a nonzero out-degree, then there is an edge (w, w_1) from w to w_1. w_1 has to be a vertex in P, for otherwise, P is not the longest directed path from v. On the other hand, if w_1 is in P, then $G_c(\widehat{X_3})$ cannot be a tree- hence w has zero out-degree. By renumbering the vertices suitably, which is equivalent to a permutation similarity of X_3, we can assume that $w = s'$, which means that X_3 is of the form

$$X_3 = \begin{bmatrix} X_4 & 0_{(s'-1)\times 1} \\ u^T & 1 \end{bmatrix} \tag{4.42}$$

Here u is a $(s' - 1) \times 1$ vector and X_4 is a $(s' - 1) \times (s' - 1)$ matrix. We now use a property of P–matrices, proved in [65] : Let M be a square matrix of the form

$$\begin{bmatrix} M_1 & a_1 \\ a_2^T & a \end{bmatrix} \tag{4.43}$$

where M_1 is a square matrix, a is a scalar and a_1 and a_2 are column vectors of suitable dimensions. Then M is a P-matrix if and only if M_1, $[a]$, and $M_1 - a_1(a^{-1})a_2^T$ are P–matrices.

Applying the above property to X_3, it is clear that X_3 is a P-matrix if and only if X_4, 1, and $X_4 - 0(1)u^T = X_4$ are P–matrices. X_4 corresponds to the digraph $G_c(X_3) - s'$ and is connected since s' has out-degree 0 and is, hence, clearly a rooted tree. By the induction hypothesis, X_4 is a P-matrix. This proves that X_3 is a P-matrix. $\square$

4.11 Conclusion

In this chapter, the importance of 2-MITE networks is shown by the fact that the D-stability of these networks is guaranteed if the input-connectivity matrix is nonsingular and has a positive diagonal. A graph-theoretic approach to the problem of synthesis using 2-MITEs is taken. An expression for the powers obtained in a 2-MITE POPL network is arrived at using the theory of Coates graphs and is shown in terms of distances between two vertices in a digraph. This leads to necessary conditions and, for the single-output case, sufficient conditions for a power matrix to be 2-MITEable.

CHAPTER 5

SYNTHESIS OF MITE LOG-DOMAIN FILTERS WITH UNIQUE OPERATING POINTS

Practical log-domain filter circuits might have multiple operating points in regions where the translinear element does not obey the exponential law. In this chapter, a method is proposed to implement any filter by a log-domain circuit that necessarily has a unique operating point. Any state-space description of the filter is shown to have an equivalent description that can be implemented by such a circuit. This methodology is applied to the synthesis of MITE filters. As an example, shifted-companion-form (SCF) filters are synthesized. Further, it is proved that the resulting filters have a unique operating point.

5.1 Introduction

Log-domain filters are usually designed under the assumption that the translinear element has ideal exponential characteristics. However, this exponential characteristic is valid only in a certain region of operation of the translinear element. Hence, though the ideal equations indicate that the circuit has a unique operating point, it might happen that the filter implementation leads to multiple operating points. The existence of multiple operating points in log-domain filters using MOSFETs in the subthreshold region is reported in [66]. However, no general procedure is known to synthesize log-domain filters in a manner that avoids this phenomenon. Here, a synthesis methodology using first-order low-pass filters, FOLPFs for short, is proposed. Synthesis using FOLPFs has the advantage that the exponential state-space transformation is already implicitly done in the FOLPF. Further, it will be shown that the state-space decomposition can be done such that the resulting circuit has a unique operating point.

5.2 Mathematical Preliminaries

The *sign pattern* of a real matrix A, denoted by $\text{sign}(A)$, is defined as the matrix obtained by replacing each element of A by its sign; i.e.,

$$[\text{sign}(A)]_{ij} = \begin{cases} 1 & \text{if } A_{ij} > 0, \\ -1 & \text{if } A_{ij} < 0, \\ 0 & \text{if } A_{ij} = 0. \end{cases}$$

The *qualitative class* $\mathcal{Q}(A)$ of a real matrix $A \in \mathbb{R}^{n\times m}$ is defined by $\mathcal{Q}(A) = \{B \in \mathbb{R}^{n\times m} | \,\text{sign}(B) = \text{sign}(A)\}$. A square matrix A is a *sign-nonsingular* (SNS) matrix if every matrix in its qualitative class is nonsingular.

5.3 Constraints on the State-Space Equations

The general state-space form of any multiple-input multiple-output (MIMO) filter is given by

$$\begin{aligned} \dot{\mathbf{x}}(t) &= A\mathbf{x}(t) + B\mathbf{u}(t) \\ \mathbf{y}(t) &= C\mathbf{x}(t) + D\mathbf{u}(t), \end{aligned} \tag{5.1}$$

where $\mathbf{x}(t) \in \mathbb{R}^n$, $\mathbf{u}(t) \in \mathbb{R}^m$, $\mathbf{y}(t) \in \mathbb{R}^p$, and A, B, C, and D are matrices of appropriate dimensions.

Definition 5.3.1 *The state-space system in Equation* (5.1) *is said to be* implementable by FOLPFs *if A has negative diagonal entries.*

Clearly, this means that one can write Equation (5.1) in terms of low-pass filters as

$$\begin{aligned} \dot{\mathbf{x}} + E\mathbf{x} &= A'\mathbf{x} + B\mathbf{u} \\ \mathbf{y} &= C\mathbf{x} + D\mathbf{u}, \end{aligned} \tag{5.2}$$

where E is a diagonal matrix with positive diagonal and $A' = A + E$ has zero diagonal.

Definition 5.3.2 *The state-space system in Equation* (5.1) *is said to have a* sign-unique operating point *if A is a SNS matrix.*

The motivation behind the above definition is seen in Theorem 1 in [36], a slightly modified version of which is the following:

Theorem 5.3.1 *Let U be an open convex subset of $\mathbb{R}^n$ and $f : U \subseteq \mathbb{R}^n \mapsto \mathbb{R}^n$ a C^1 function such that all the elements of the Jacobian matrix $Df(\mathbf{x})$ of f have the same sign for all $\mathbf{x} \in U$. Then, f is injective on U if $Df(\mathbf{x})$ is a SNS matrix.*

It will be seen that solving for the operating point of a Multiple-Input Translinear Element (MITE) implementation of Equation (5.1) requires the solution of a nonlinear equation of the form $f(\mathbf{V}) = 0$, where $f : (0, V_{\text{DD}})^n \mapsto \mathbb{R}^n$ is such that the partial derivative $\frac{\partial f_i}{\partial x_j}$ has the same sign as A_{ij}. Hence, the operating point is unique if A is a SNS matrix. Therefore, all state-space systems will be required to have a sign-unique operating point.

5.3.0.1 Example: Shifted-Companion-Form Filters

The shifted-companion-form (SCF) [67] lends itself easily to synthesis by the proposed methodology. The MITE implementation of a SCF state-space system is particularly simple in the case where the transmission zeros are formed by summation of the state variables. The (A, B, C, D) matrices from Equation (5.1) of this single-input single-output system are [67]

$$A = \begin{bmatrix} -a_{n-1} - \alpha & -a_{n-2} & -a_{n-3} & \cdots & -a_1 & -a_0 \\ 1 & -\alpha & 0 & \cdots & 0 & 0 \\ 0 & 1 & -\alpha & \cdots & 0 & 0 \\ & & & \vdots & & \\ 0 & 0 & 0 & \cdots & 1 & -\alpha \end{bmatrix}, \tag{5.3}$$

$B = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^t$, $C = \begin{bmatrix} b_{n-1} & b_{n-2} & \cdots & b_0 \end{bmatrix}$, and $D = d$ (a scalar).

The above state-space realization is obtained from the transfer function

$$\frac{Y(s)}{U(s)} = \frac{b_{n-1}(s+\alpha)^{n-1} + b_{n-2}(s+\alpha)^{n-2} + \cdots b_0}{(s+\alpha)^n + a_{n-1}(s+\alpha)^{n-1} + \cdots a_0} + d$$

Theorem 5.3.2 *A is a SNS matrix if $\alpha > 0$ and $a_0, a_1, \ldots, a_{n-1}$ are nonnegative. If $\alpha = 0$, then the companion matrix A is a SNS matrix if $a_0 \neq 0$.*

Proof : Let $\alpha > 0$ and $a_0, a_1, \ldots, a_{n-1}$ be nonnegative. It suffices to show that $M\mathbf{x} = 0 \Rightarrow \mathbf{x} = 0$ for any $M \in \mathcal{Q}(A)$. $M\mathbf{x} = 0$ implies that

$$\begin{aligned}
&|M_{nn-1}|x_{n-1} = |\mathcal{M}_{nn}|x_n \\
&|M_{n-1n-2}|x_{n-2} = |\mathcal{M}_{n-1n-1}|x_{n-1} \\
&\qquad\qquad \vdots \\
&|M_{21}|x_1 = |\mathcal{M}_{22}|x_2 \\
&|M_{11}|x_1 + |\mathcal{M}_{12}|x_2 + \cdots |\mathcal{M}_{1n}|x_n = 0
\end{aligned} \tag{5.4}$$

It should be noted that all the elements of M above (except the last row) and M_{11} are necessarily nonzero. It is clear that $x_i = \beta_i x_n$ for $i \in [1:n]$ with $\beta_i > 0$. The last equation in Equation (5.4) yields $x_n \sum_{i=1}^{n} |M_{1i}|\beta_i = 0$, which implies that x_n and hence $\mathbf{x}$ is zero. The proof is similar and easier when $\alpha = 0$. $\square$

To show that constraining A in Equation (5.1) to be a SNS matrix does not restrict the set of transfer functions obtainable from Equation (5.1), the following result is proved:

Theorem 5.3.3 *There exists a SNS matrix J with negative diagonal entries similar to any Hurwitz matrix A. In particular, J can be written as a direct sum, i.e., a block diagonal matrix, of shifted-companion matrices of the type shown in Equation* (5.3) *with* $\alpha > 0$.

Proof : J can be chosen to be the real Jordan canonical form of A [28]. A more useful SNS matrix similar to a Hurwitz matrix A is obtained by the following method:
Let $\alpha > 0$ be such that $-\alpha > \max_{\lambda \in \sigma(A)} \Re e(\lambda)$, where $\sigma(A)$ is the set of eigenvalues of A. Define $A' = A + \alpha I$. Clearly, $\sigma(A') = \{\lambda + \alpha | \lambda \in \sigma(A)\}$ and hence, by the definition of α, A' is Hurwitz. Taking the rational form or the rational canonical form of A' [28], a matrix J' is obtained that is a direct sum of companion matrices of the form depicted in Equation (5.3) with $\alpha = 0$. It follows from the assumption that A' is Hurwitz that each block companion matrix in J' is Hurwitz. Hence, the first row of each block is nonnegative. If $A' = S^{-1}J'S$, then $A = A' - \alpha I = S^{-1}(J' - \alpha I)S$. Clearly, $J = J' - \alpha I$ is a direct sum of shifted-companion matrices each of which satisfies the conditions of Theorem 5.3.2 so that J is a SNS matrix with negative diagonal entries. $\square$

Because of the above theorem, it can be assumed that the given state-space system in Equation (5.1) has a sign-unique operating point and is implementable by FOLPFs. A synthesis procedure for implementing such a state-space system is given below:

5.4 Synthesis Procedure

Step 5.4.1 (Dimensionalization) *The variables will be first scaled [18] so that each signal is replaced by a ratio of a signal current to a unit current which gets canceled out as the system is linear. The derivative* $\frac{\mathrm{d}}{\mathrm{d}t}$ *is replaced by* $\tau\frac{\mathrm{d}}{\mathrm{d}t}$. *Hence, each state-space equation can be written as:*

$$\tau\frac{\mathrm{d}I_{x_i}}{\mathrm{d}t} + E_i I_{x_i} = \sum_{j=1}^{n} A'_{ij} I_{x_j} + \sum_{k=1}^{m} B_{ik} I_{u_k} \;\; i \in [1:n] \tag{5.5}$$

$$I_{y_i} = \sum_{j=1}^{n} C_{ij} I_{x_j} + \sum_{k=1}^{m} D_{ik} I_{u_k} \quad i \in [1:p] \tag{5.6}$$

Step 5.4.2 (FOLPF implementation) *A MITE FOLPF [16, 18] used in the* i^{th} *first-order equation in Equation* (5.5) *is shown in Figure 5.1(a). The MITE network satisfies the equation:*

$$\frac{CU_T}{\kappa}\frac{\mathrm{d}I_{x_i}}{\mathrm{d}t} + I_{\tau_i} I_{x_i} = I'_{\tau_i} I_{in_i} \tag{5.7}$$

Fix a value of C *and define* $I_\tau = \frac{CU_T}{\kappa\tau}$. *Define* $I_{\tau_i} = E_i I_\tau$. *Choose a* $\alpha_i > 0$, *typically the magnitudes of one of the coefficients in the right hand side of Equation* (5.5)*), and define* $I'_{\tau_i} = \alpha_i I_\tau$. *Hence, the required input current* I_{in_i} *to the filter is given by* $I_{in_i} = \sum_{j=1}^{n} \frac{A'_{ij}}{\alpha_i} I_{x_j} + \sum_{k=1}^{m} \frac{B_{ik}}{\alpha_i} I_{u_k}$.

Step 5.4.3 (Multiplier implementation) *The multiplications* $(\frac{A'_{ij}}{\alpha_i})I_{x_j}$, $(\frac{B_{ik}}{\alpha_i})I_{u_k}$, $C_{ij}I_{x_j}$, *and* $D_{ik}I_{u_k}$ *in Equation* (5.5) *and Equation* (5.6) *are implemented through straightforward methods given in [18]. The inputs* I_{u_k} *are passed through diode-connected MITEs to generate the voltages* V_{u_k} *as shown in Figure 5.1(b). Hence,* I_{x_i} *is associated with a voltage* V_{x_i} *at the output MITE of the* i^{th} *FOLPF shown in Figure 5.1(a) and similarly,* I_{u_k} *is associated with* V_{u_k}. *The circuits for* $(\frac{A'_{ij}}{\alpha_i})I_{x_j}$, $(\frac{B_{ik}}{\alpha_i})I_{u_k}$, *shown respectively in Figure 5.1(c)* *and Figure 5.1(d), are in terms of these voltages. The products for the output currents are generated in an identical fashion.*

Step 5.4.4 (Summation) *The inputs I_{in_i} to the FOLPFs and the outputs I_{y_i} are found simply by using KCL, through a current mirror if needed as shown in Figure 5.1(c) and Figure 5.1(d). Also, the output MITE of each FOLPF can be removed unless the state variable I_{x_i} is itself one of the output currents I_{y_i}. Consolidation [18] can be used to remove redundancies whenever possible.*

5.4.0.2 Example : SCF filter synthesis

For $\alpha > 0$, the SCF state-space equations are implementable by FOLPFs and have a sign-unique operating point. Though the synthesis procedure detailed above can be used directly, a convenient scaling of the state variables before applying the synthesis procedure results in a much simpler topology. Define $T = \mathrm{diag}(1, a_{n-2}, a_{n-3}, \ldots, a_0)$. The SCF system in Section 5.3.0.1 is transformed according to $A' = TAT^{-1}$, $B' = TB$, $C' = CT^{-1}$, and $D' = D$. The modified system is given by the following equations:

$$\dot{x}_1 + (\alpha + a_{n-1})x_1 = u - x_2 - x_3 \cdots - x_n$$

$$\dot{x}_2 + \alpha x_2 = a_{n-2} x_1$$

$$\dot{x}_3 + \alpha x_3 = \frac{a_{n-3}}{a_{n-2}} x_2$$

$$\vdots$$

$$\dot{x}_n + \alpha x_n = \frac{a_0}{a_1} x_{n-1}$$

$$y = b_{n-1}x_1 + \frac{b_{n-2}}{a_{n-2}}x_2 + \frac{b_{n-3}}{a_{n-3}}x_3 + \cdots \frac{b_0}{a_0}x_n + du$$

It should be noted that the state variable equations are a cascade of FOLPFs with input $u - x_2 - x_3 \cdots - x_n$. Consolidation can be applied to a cascade of FOLPFs since the output MITE of a FOLPF and the input MITE of the FOLPF following it can be removed and the corresponding voltages connected, as shown in Figure 5.2. The whole generic SCF filter is shown in Figure 5.2. Note that the required multiplier blocks are easily synthesized as described in the synthesis procedure. Also, this block can be used as a "universal active filter" to generate filters of any type and any order. For those filters that do not pass DC, an offset needs to be applied at the output so that the requirement of positive currents through the MITEs is satisfied.

5.5 Uniqueness of the Operating Point

For determining conditions on the synthesized filter such that the operating point is unique, a general model for a MITE that covers all regions of operation of the basic translinear element (BJT, MOSFET etc.) constituting the MITE is needed. Based on a model for a MITE that assumes the weighted sum of voltages to be ideal, sufficient conditions on the MITE network topology for the operating point to be unique have been given elsewhere [51]. As only 2-MITEs are required for this synthesis methodology, even the requirement of ideal weighted summation can be discarded. For the nonideal model of the MITE in Figure 5.3, the current through the input gates will be required to zero. The drain current is assumed to be of the form $I = h(V_1, V_2, V_{\mathrm{d}})$, where $h : (0, V_{\mathrm{DD}})^3 \mapsto (0, \infty)$ is a C^1 map satisfying : $\forall\, (V_1, V_2, V_{\mathrm{d}}) \in (0, V_{\mathrm{DD}})^3$

$$
\begin{aligned}
\text{Transconductance 1} \quad g_{\mathrm{m}_1} &\triangleq \frac{\partial h}{\partial V_1} > 0, \\
\text{Transconductance 2} \quad g_{\mathrm{m}_2} &\triangleq \frac{\partial h}{\partial V_2} > 0, \\
\text{Output conductance} \quad g_o \ \ &\triangleq \frac{\partial h}{\partial V_{\mathrm{d}}} \geq 0
\end{aligned}
\tag{5.8}
$$

In a floating-gate implementation, this is nothing more than the assumption that the nonideal weighted summation is monotonically increasing along with the requirements of positive transconductance and nonnegative output conductance of the MOSFET. A similar form can be given for the drain current through a PFET in a current mirror or a current source, taking care about the signs for the different conductances. A brief proof of the uniqueness of the operating point is as follows:

Theorem 5.5.1 *The DC MITE circuit realizing Equation* (5.1) *according to the synthesis procedure in Section 5.4 has at most one operating point with the operating point voltages in* $(0, V_{\mathrm{DD}})$ *if* A *is a SNS matrix.*

Proof : Since all the elements (MITEs or PFETs) are voltage controlled, the nonlinear node equation [27]: $f(\mathbf{V}) = 0$ can be written, where $f : (0, V_{\mathrm{DD}})^k \mapsto \mathbb{R}^k$, $\mathbf{V}$ being the vector of drain-to-ground voltages of the MITEs with common drains. It should be noted that the MITEs that are the outputs of products in Equation (5.6) do not affect the operating point uniqueness. By Theorem 5.3.1, it suffices to prove that the Jacobian $Df(\mathbf{V})$ does not change sign patterns for $\mathbf{V} \in (0, V_{\mathrm{DD}})^n$ and that Df is a SNS matrix. It can be seen from the way the MITEs are connected that $(Df(\mathbf{V}))_{ij}$ has a sign independent of $\mathbf{V}$. Df is nothing but the node-admittance matrix of the linear network N obtained by setting the DC sources to zero and replacing the nonlinear elements (PFETs or MITEs) by their small-signal equivalent circuits according to Equation (5.8). Consider the set $\mathcal{N}$ of linear networks obtained by changing only the magnitude of different transconductances and conductances in N. Any matrix $M \in \mathcal{Q}(Df)$ is obtained as the node-admittance matrix of an element N' of $\mathcal{N}$. To show that $\det(M) \neq 0$, it suffices to prove that the corresponding network has a unique solution in which the node-to-ground voltages $\mathbf{V}$ are zero. The following easily provable observations about voltages in N', which correspond to the voltages in Figure 5.1, are made:

1. The voltages V_{u_k}, V'_{ij}, V_{ik}, and V_{ref} are zero.
2. Wherever the voltage V_{in_i} appears in the linear node equations, it can be replaced by $\beta_{ii} V_{x_i}$ where $\beta_{ii} > 0$.
3. If $K = \mathrm{sign}(A')$, then for some arbitrary $\beta_{ij} > 0$, $V_{\mathrm{in}_i} = -\sum_{\substack{j=1 \\ j \neq i}}^{n} \beta_{ij} K_{ij} V_{x_j}$

Combining the results in 2) and 3), it is clear that $L\mathbf{V} = 0$ where $V = (V_{x_1}, V_{x_2}, \ldots, V_{x_n})^t$ and $L \in \mathbb{R}^{n \times n}$ is given by $L_{ii} = \beta_{ii}$ and for $i \neq j$, $L_{ij} = \beta_{ij} K_{ij}$. By the definition of K, L is in $\mathcal{Q}(A)$. Since A is a SNS matrix, $det(L) \neq 0$ and hence $\mathbf{V} = 0$, which implies that all the node-to-ground voltages in N' are zero. $\square$

5.6 Conclusion

Conditions on the state-space equations for log-domain filters that ensure the uniqueness of the operating point of the resulting circuit have been presented. A synthesis procedure using first-order low-pass filters to implement any log-domain filter has been described. It is proved that the operating point for the synthesized filter is unique. As an example, shifted-companion-form filters of arbitrary type and order are synthesized.

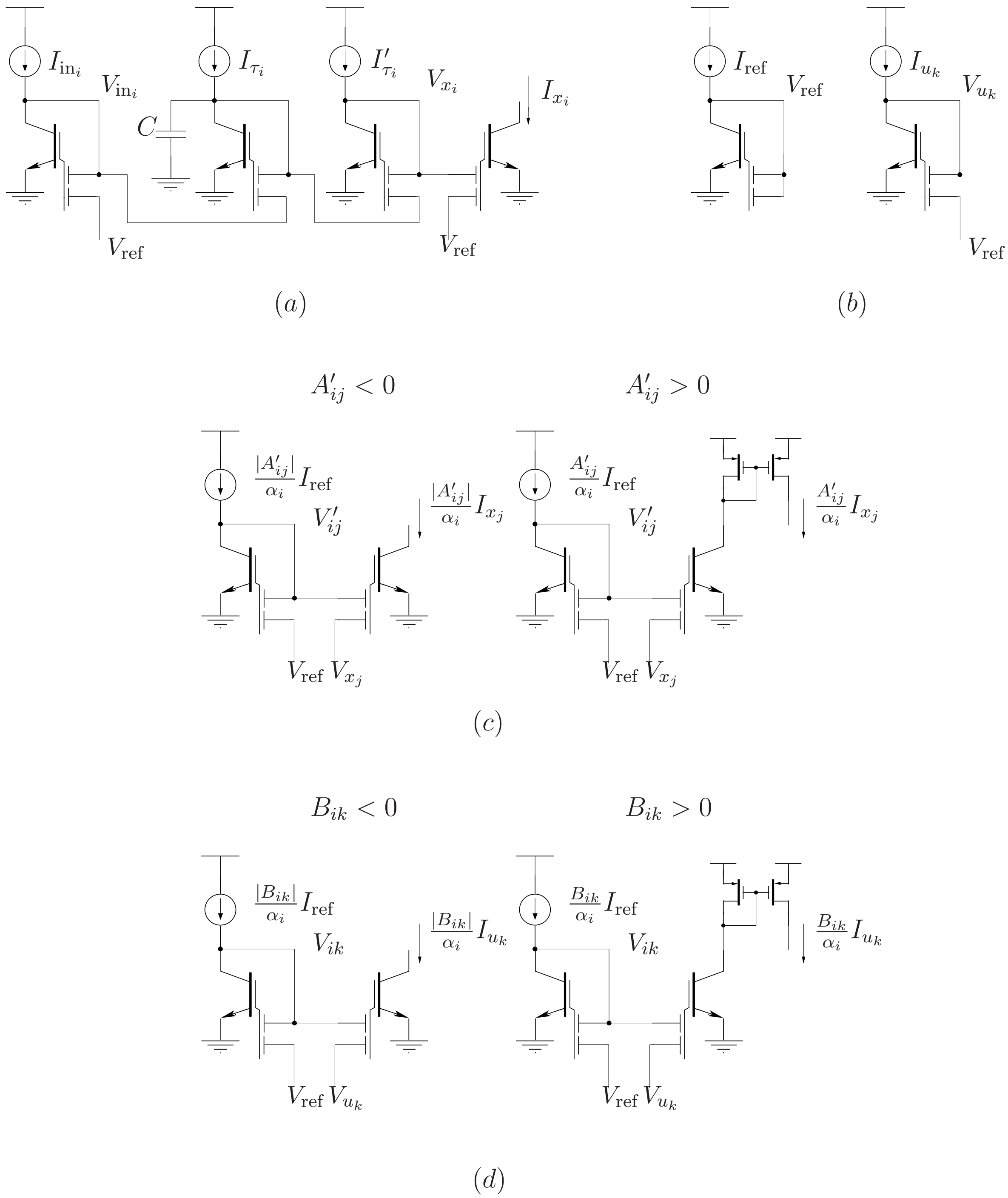


Figure 5.1. The circuit blocks used in implementing the first-order equations in Equation (5.5). **(a) The MITE low-pass filter used in the** i^{th} **equation in Equation** (5.5). **(b) The MITE circuits for generating the voltages** V_{ref} **and** V_{u_k} **used in the multipliers. (c) The MITE circuits implementing the product** $(\frac{A'_{ij}}{\alpha_i})I_{x_j}$ **for** $A'_{ij} > 0$ **and** $A'_{ij} < 0$. **(d) The MITE circuits implementing the product** $(\frac{B_{ik}}{\alpha_i})I_{u_k}$ **for** $B_{ik} > 0$ **and** $B_{ik} < 0$.

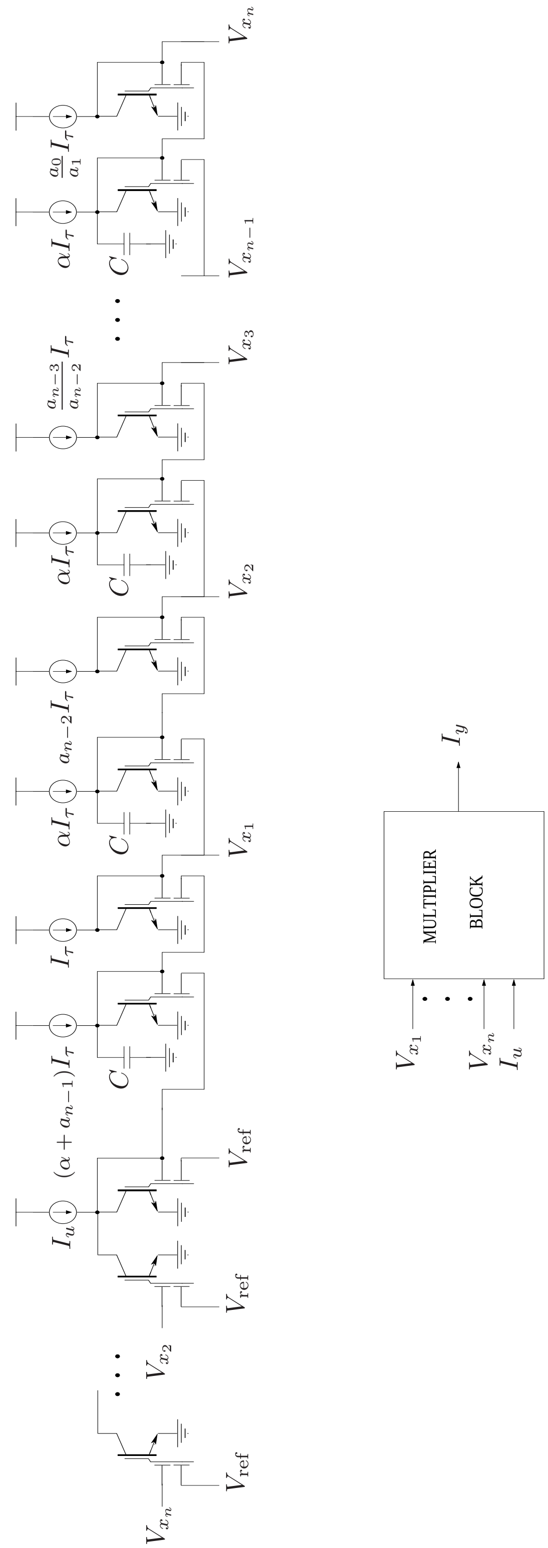


Figure 5.2. A generic shifted-companion-form filter. It can be used to generate any linear transfer function. The multiplier block is implemented as described in Section 5.4.

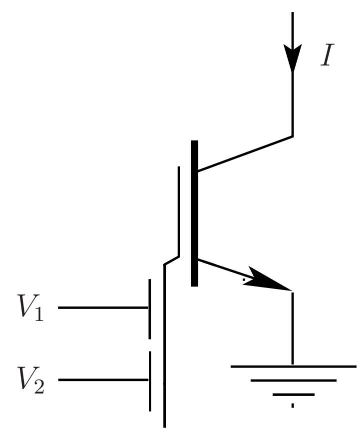


Figure 5.3. Symbol for a 2-MITE. Ideally, it should obey the law $I = I_\mathrm{s} \exp(\frac{\kappa}{U_\mathbf{T}}(V_1 + V_2))$, where I_s is a scaling current.

CHAPTER 6
SYNTHESIS EXAMPLES

6.1 Synthesis of static functions

The implementation of nonlinear functions using translinear circuits is discussed in [4]. Elementary operations like addition and subtraction are easily performed in any current-mode system by using Kirchoff's current law (KCL) and a current mirror, respectively. Translinear circuits, in particular, can also do other elementary operations like multiplication, division, and exponentiation with rational exponents. Hence any algebraic function can be synthesized by simply expressing it in terms of these elementary operations. Transcendental functions like $\exp(x), \log(x), \arctan(x)$ are implemented by suitably approximating them using algebraic functions. Different techniques exist for approximating functions by rational functions [68], approximation with minimax or near-minimax error being one of the more suitable methods for approximation over an interval. Remez's algorithm [68] is used to determine the minimax rational approximation while numerous other techniques exist to get near-minimax approximations. For example, Maple's 'minimax' command implements Remez's algorithm.

6.1.1 Current Splitters

The basic translinear element in a translinear circuit (MITE, BJT, etc.,) usually accepts currents of only one sign. If it is known that a bidirectional current is bounded below by some known value, then it is easy to see that simply by providing the required offset to the bidirectional current, one can convert it to a current of one sign alone. However, this is of no use when the bounds on the signal are unknown. Even if the bounds are known but are large in magnitude, the required power consumption is sometimes prohibitive. Current splitters circumvent this by "splitting" I as two positive currents I_p and I_n (i.e., $I = I_p - I_n$) that remain positive no matter what the value of I is. The two popular current splitters are (a) the geometric current splitter, where $I_p I_n = I_b^2$ for some constant current I_b, and (b) the harmonic current splitter, where $I_p I_n/(I_p + I_n) = I_b$. It can shown by solving for I_p and I_n in terms of I that these currents remain positive for any value of I. However, the currents in a harmonic splitter always remain greater than I_b, a fixed positive value, though they get arbitrarily close to it as $|I| \to \infty$. The geometric splitter currents are not bounded below by any positive current.

Though the concept of the geometric current splitter seems to be a natural consequence of the equations in a BJT Class AB output stage, [69] seems to be the first publication of the use of the harmonic mean output stage, although the harmonic mean circuit itself was derived in [4]. The use of the current splitters in Class AB log–domain filtering is given in [70, 71, 8, 72, 73, 74].

6.1.1.1 MITE Geometric current splitter

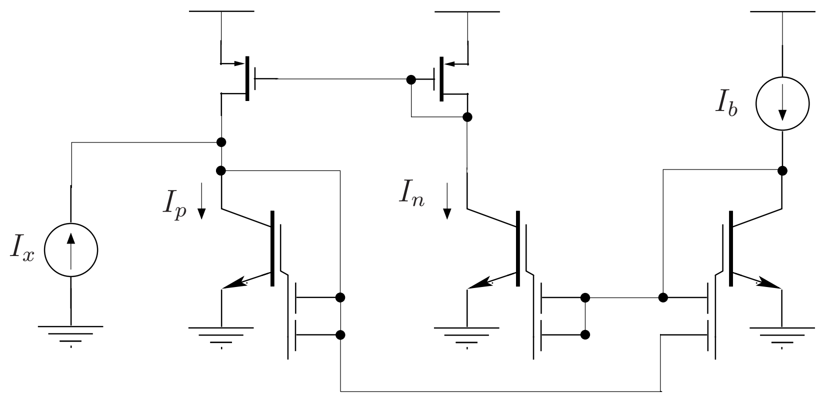
The MITE implementation of the geometric constraint $I_p I_n = I_b^2$, or equivalently, the translinear loop equation $I_b^2 I_p^{-1} I_n^{-1} = 1$, is shown in Figure 6.1(a) along with the necessary current mirroring to implement $I_p - I_n = I_x$, where I_x is the bidirectional input current. An alternative implementation, that is useful in some applications to be discussed later, is shown in Figure 6.1(b). A plot of I_p and I_n, obtained using the models for a AMI .5μ process, is shown at Figure 6.3 for $I_b = 10$nA.

6.1.2 Particle filters and target tracking

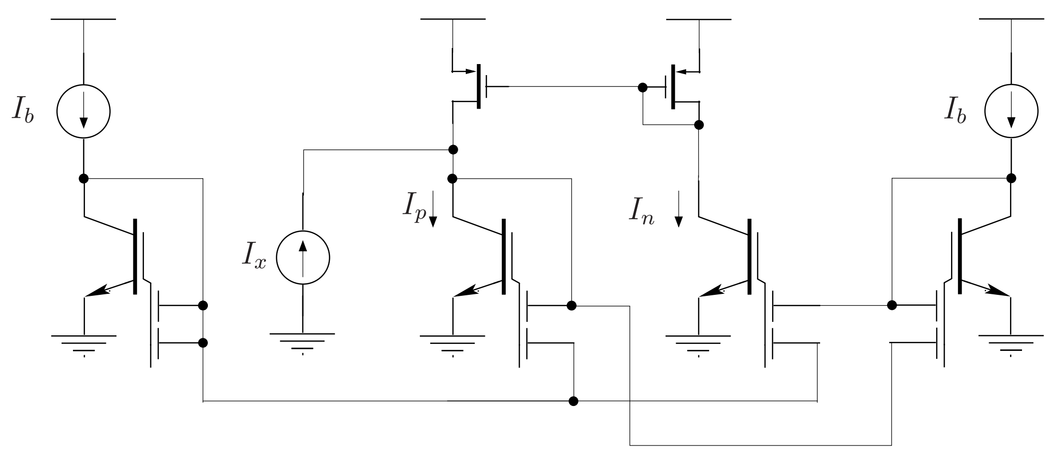
Particle filters are a class of recursive simulation methods used for estimating the state of a discrete-time system in the presence of noise from a set of observations made on the system. The state-space models can be nonlinear. Bearings-only tracking involves estimating the target states based upon angle measurements. The particle filter algorithm, in this context, involves the calculation of the following function [75, 76]:

$$\frac{1}{\sqrt{2\pi{\sigma_r}^2}} \exp - \frac{\left(z_k - \arctan \frac{y_k^{(i)}}{x_k^{(i)}}\right)^2}{2{\sigma_r}^2}, \tag{6.1}$$

where z_k is the angle measurement obtained at a suitable sensor node and x_k^i, y_k^i are components of the i^{th} proposed particle. The angle measurement may be obtainable directly in analog [76]. As a result of the collaborative work done with Dr. Rajbabu Velmurugan and Dr. James McClellan of the Center for Signal and Image Processing at Georgia Tech, we explore the possibility of implementing all or part of the particle filtering algorithm using MITEs.



(a)



(b)

Figure 6.1. Two MITE circuits for the geometric current splitter. The current I_x is split into two currents I_p and I_n such that their geometric mean is a fixed bias current I_b.

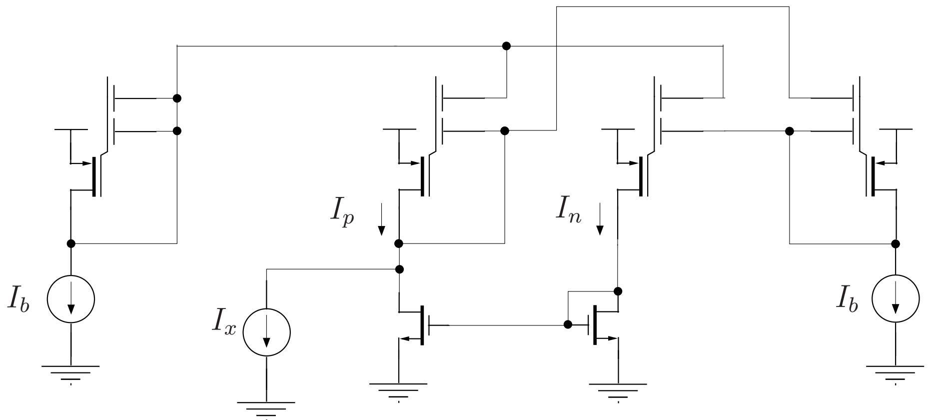


Figure 6.2. The version of the circuit in Figure 6.1(b) with PFET floating-gate MOSFETs for the MITEs.

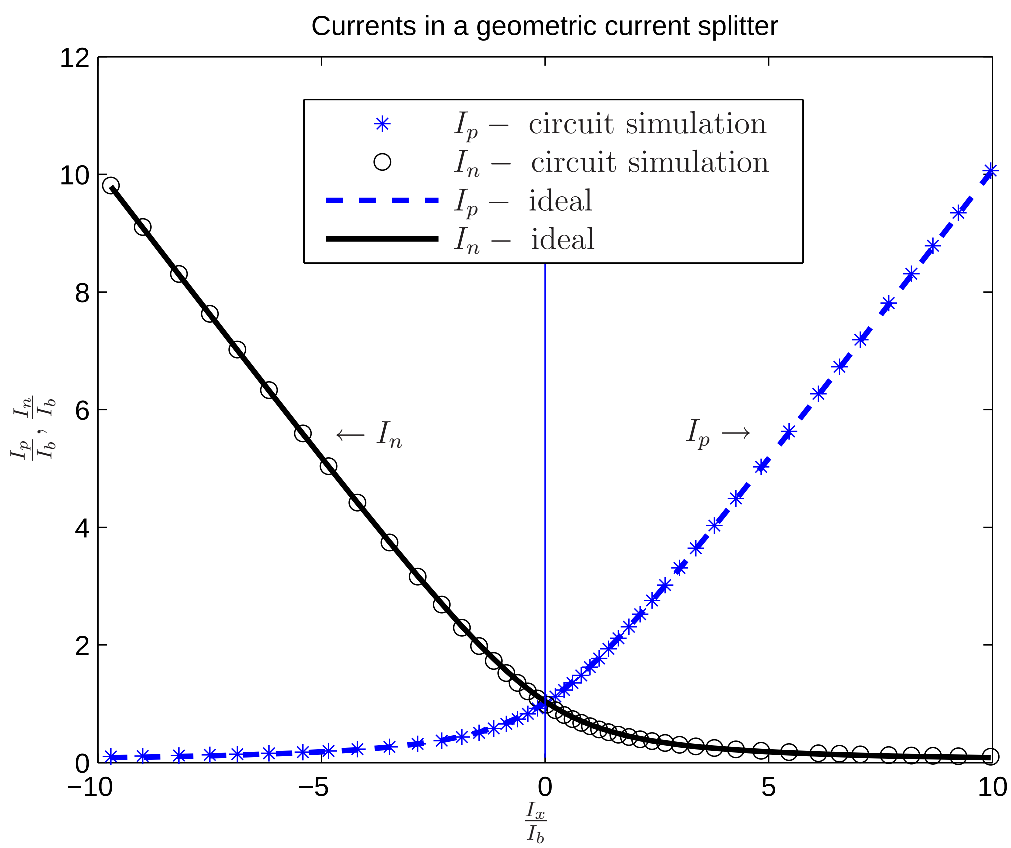


Figure 6.3. Simulation of the currents in a MITE geometric current splitter for the circuit shown in Figure 6.2.The value of I_b is 10nA.

The transcendental functions that need to be implemented in the bearings-only tracking algorithm are the inverse tangent $\arctan(x)$ and the Gaussian $\exp(-x^2/2)$. The approximations and the corresponding implementations of these functions are considered below.

6.1.3 Implementing static functions with a geometric current splitter

Let $y = f(x)$ be the desired functional behavior of some translinear block and also suppose that x can take values of both signs. As discussed before, there are many advantages to using a current splitter. However, the output of a current splitter is two currents and hence it might be thought that the implementation would be considerably more complicated than if we had just a single positive input. It is our aim here to show that this is not the case, especially when f is an odd or even function. Throughout this section, a geometric splitter is used; the results do not automatically translate to other cases.

First, we split f into its odd and even parts, i.e., we define $f_e(x) = (f(x) + f(-x))/2$ and $f_o(x) = (f(x) - f(-x))/2$. Since f_e and f_o themselves are even and odd, respectively, we can assume that $f_e(x) = g(x^2)$ and that $f_o(x) = xh(x^2)$. Let x_+ and x_- be the outputs of the current splitter such that $x_+x_- = a^2$ for some constant a. We have $x^2 = x_+^2 + x_-^2 - 2x_+x_- = rb - a^2$, where the new variable r is defined as $r = (x_+^2 + x_-^2)/b$. If we define $g(x^2) = g(rb - a^2) = \tilde{g}(r)$ and $h(x^2) = h(rb - a^2) = \tilde{h}(r)$, then it is clear that $f(x) = \tilde{g}(r) + (x_+ - x_-)\tilde{h}(r)$. Hence, the majority of the computation is in terms of the new *positive* variable r: x_+ and x_- themselves separately enter the picture only through one multiplication. Using this procedure the process of computing nonlinear functions is, therefore, considerably simplified.

6.1.4 Implementation of the inverse tangent function

The function ϕ to be approximated is as follows (normalized so that $\phi(\infty) = 1$)

$$\phi(x) = \frac{2}{\pi}\arctan(x), \text{ where } |x| < \infty. \tag{6.2}$$

An approximation of ϕ using algebraic functions, given in [4], is as follows:

$$y = f(x) = \frac{x}{0.63 + \sqrt{0.88 + x^2}}, \text{ where } |x| < \infty. \tag{6.3}$$

The maximum error obtained using the approximation is less than 0.05% of the maximum value.

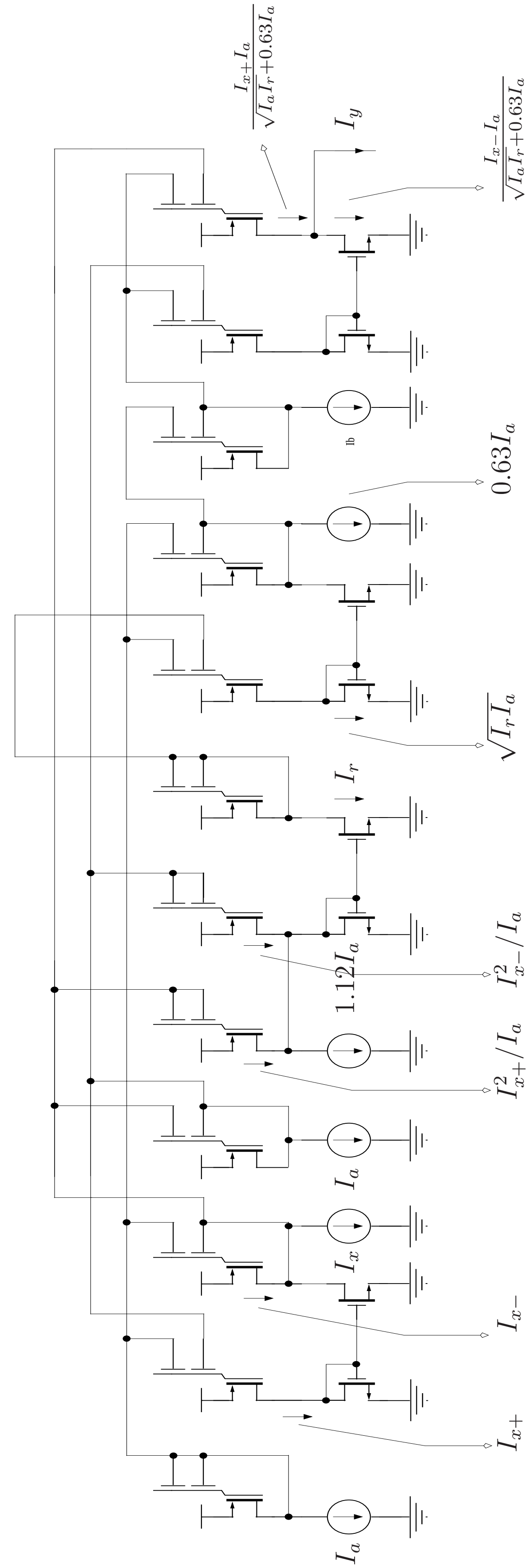


Figure 6.4. The MITE circuit implementing the arctan function. Here, $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a - 1.12I_a$ and $I_y = (I_{x+}I_a - I_{x-}I_a)/(\sqrt{I_a I_r} + 0.63I_a)$

.

The implementation of f using MITEs is done through the following steps:

1. **Scaling** Since the input and output variables are represented by currents, to maintain dimensional consistency, the substitutions $x \mapsto I_x/I_a$ and $y \mapsto I_y/I_b$ are done. Hence, we have
$$I_y = \frac{I_x I_b}{0.63I_a + \sqrt{0.88I_a^2 + I_x^2}} \tag{6.4}$$

2. **Current splitting** Since the input x can take both positive and negative values and since the currents through the MITEs must necessarily be positive, we use a geometric current splitter to produce currents I_{x+} and I_{x-} satisfying $I_{x+} - I_{x-} = I_x$ and $I_{x+}I_{x-} = I_a^2$.

3. **Block reduction** The equation to be implemented thus becomes
$$\begin{aligned} I_y &= \frac{I_{x+}I_b - I_{x-}I_b}{0.63I_a + \sqrt{-1.12I_a^2 + I_{x+}^2 + I_{x-}^2}} \\ &= \left(\frac{I_{x+}I_b}{(0.63I_a + (\sqrt{I_r I_a}))}\right) - \left(\frac{I_{x-}I_b}{(0.63I_a + (\sqrt{I_r I_a}))}\right), \end{aligned}$$
where $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a - 1.12I_a$. The parentheses show the order in which the operations are implemented. Each block, which represents an operation in the parentheses, is implemented using procedures described in the thesis. In this case, however, since the calculation is essentially a "cascade" of simple calculations, no particular advantage of the methods described here over the previously existing methods is seen.

4. **Consolidation** As described in [18], redundant MITEs are removed using *consolidation* and the final circuit is then obtained. The final circuit is shown in Figure 6.4.

6.1.4.1 Simulation Results

The dc simulations results, using the models of a AMI 0.5μ process, of the arctan block are shown in Table 6.1. Throughout the simulation, I_b is fixed at 10nA. The range of I_a is determined by the requirement of $10I_a$ being in the subthreshold region. The dc simulation is obtained by varying the input slowly in transient analysis, for otherwise the floating-gate capacitances will be "open" in a dc analysis. A sample plot of the transfer characteristic, given for $I_a = I_b = 10$nA is shown in Figure 6.1.4.1.

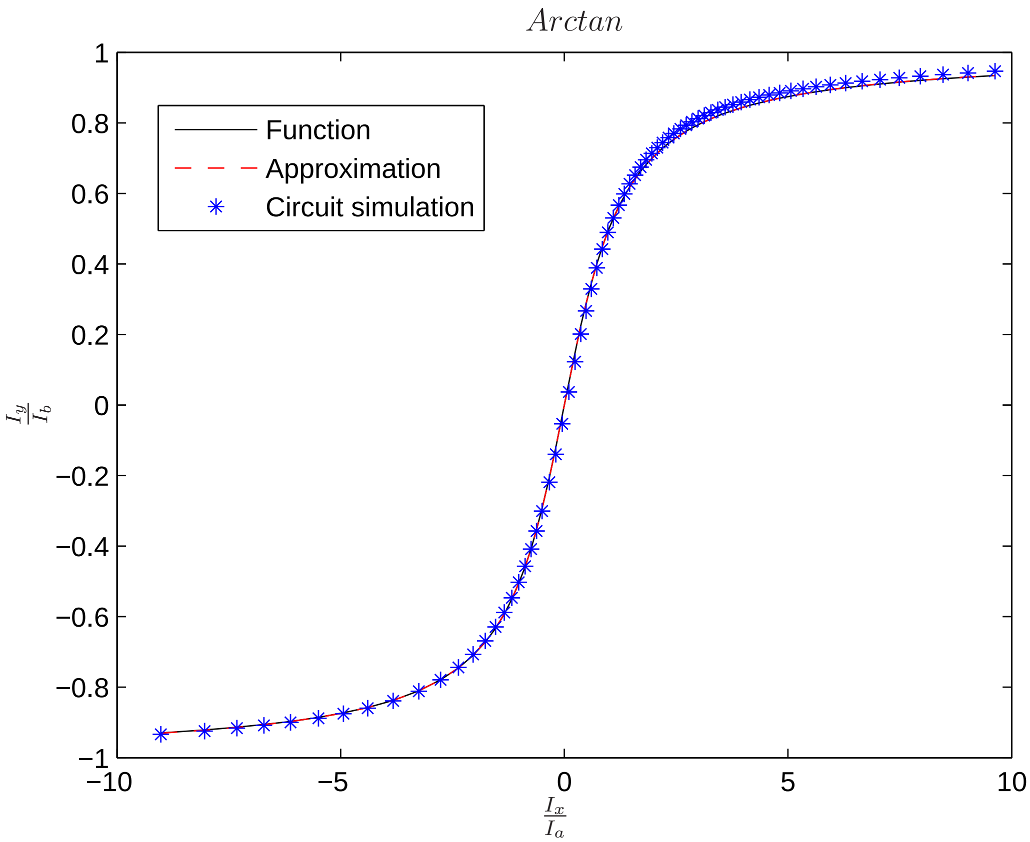


Figure 6.5. Results of simulation of the Arctan circuit. The currents I_a and I_b are each 10nA. The "function" refers to $\phi(I_x/I_a) = 2/\pi \arctan(I_x/I_a)$, the "approximation" is $f(I_x/I_a) = \frac{I_x}{0.63I_a+\sqrt{0.88I_a^2+I_x^2}}$, and the "Circuit simulation" is the simulation plot of the circuit in Figure 6.4.

Table 6.1. Simulated characteristics of the arctan circuit

Circuit	*arctan*	
	Minimum	Maximum
Ref. I_a (nA)	6	26
Input current	$-10I_a$	$10I_a$
Power (μW)	2.9	10.3
Error (%)	1.59	4.53

6.1.5 Implementation of the Gaussian

1. **Function** The Gaussian is $\phi(x) = c\exp(-x^2/(2a^2))$, where $x \mapsto I_x/I_a$, $y \mapsto I_y/I_a$, and $c \mapsto I_c/I_a$. After scaling and normalization, the Gaussian is thus transformed into $I_y = I_c\exp(-I_x^2/(2I_a^2))$.

2. **Current splitting** As before, a geometric current splitter is used satisfying $I_{x+} - I_{x-} = I_x$; $I_{x+}I_{x-} = I_a^2$.

3. **Approximation** We have $I_x^2 = I_{x+}^2 + I_{x-}^2 - 2I_a^2 = I_rI_a - 2I_a^2$, where $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a$. Thus, $I_y = eI_c\exp(-I_r/(2I_a))$. It should also be noted that if the implementation is to be valid for $I_x \in [-bI_a, bI_a]$, then it suffices to approximate $\exp(-I_r/(2I_a))$ for $I_r \in [2I_a, (2+b^2)I_a]$. The minimax rational approximation for $b = 4$ with the numerator and denominator degrees equal to 1 and 2, respectively, was found using Remez's algorithm and is given by :

$$I_y = zI_c\frac{I_a - I_{n1}(I_r/I_a)}{I_a - I_{d1}(I_r/I_a) + I_{d2}(I_r^2/I_a^2)}, \tag{6.5}$$

where $I_r = (I_{x+}^2 + I_{x-}^2)/I_a$, $I_{n1} = 0.07195I_a$, $I_{d1} = 0.2913I_a$, $I_{d2} = 0.1641I_a$, and $z = 1.245$. The maximum absolute error of this approximation in the range $I_r \in [2I_a, 18I_a]$ is 0.82% of the maximum value.

Clearly, this requires the computation of the following POPL equations: $I_{o1} = I_{n1}(I_r/I_a)$, $I_{o2} = I_{d1}(I_r/I_a)$, and $I_{o3} = I_{d2}(I_r^2/I_a^2)$. The translinear loop matrix and a solution connectivity matrix Z obtained using the method of diophantine equations described

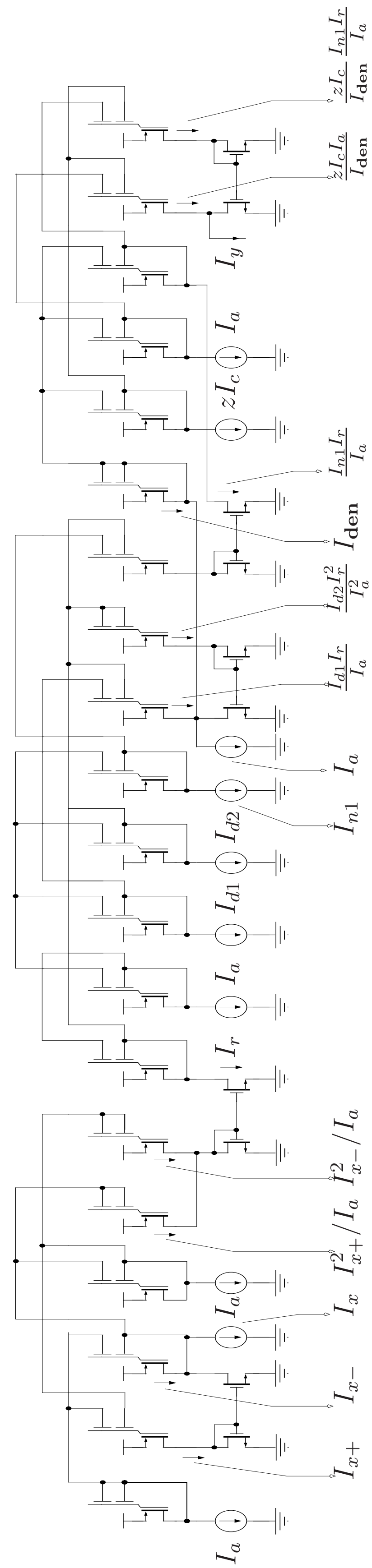


Figure 6.6. The MITE circuit to implement the Gaussian function. $I_r = I_{x+}^2/I_a + I_{x-}^2/I_a$

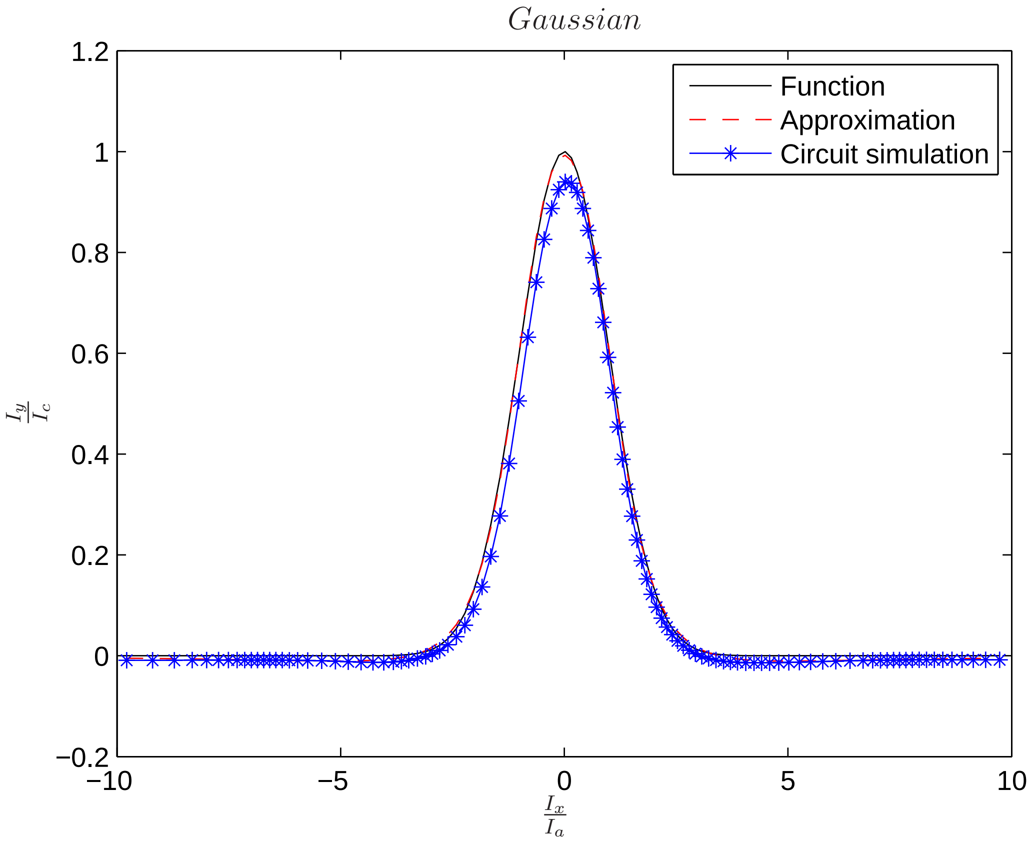


Figure 6.7. The results of simulation of the gaussian circuit. The value of I_a is 20nA and the value of I_c is 10nA.

in Chapter 3 is given below:

$$A = \begin{bmatrix} 1 & -1 & 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 0 & 0 & -1 & 0 \\ 2 & -2 & 0 & 0 & 1 & 0 & 0 & -1 \end{bmatrix}; \; Z = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 2 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 2 & 0 & 0 & 0 & 0 \end{bmatrix}$$

4. The currents I_{n1}, I_{d1}, and I_{d2} are set using programmable floating–gate MOSFETs [77]. The final circuit is shown in Figure 6.6. Simulation results are shown in Figure 6.1.5.

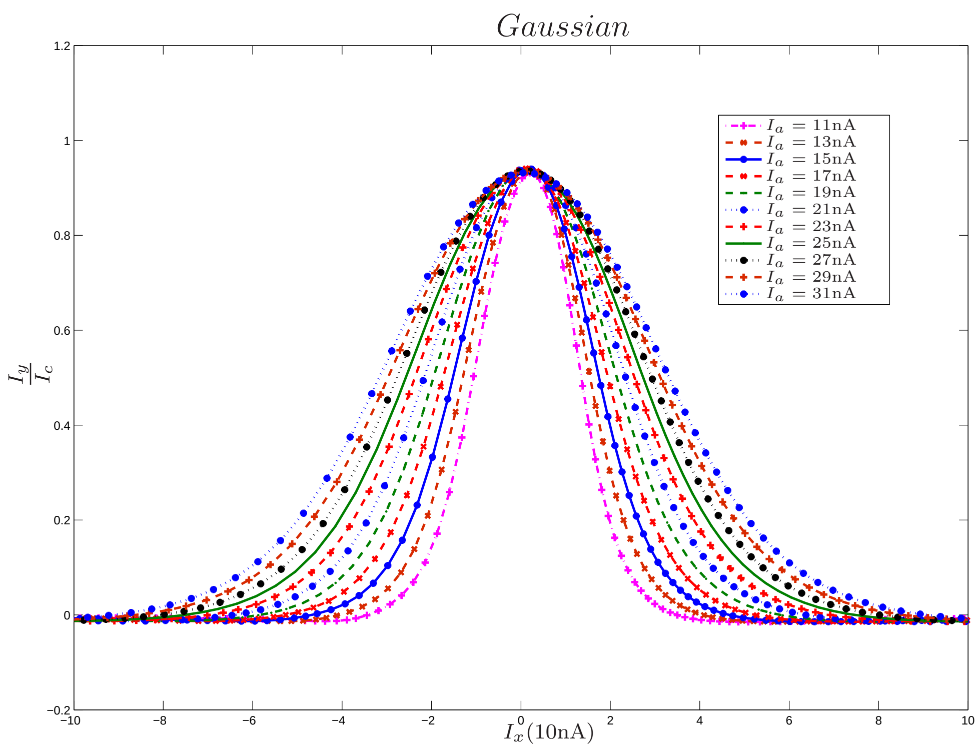


Figure 6.8. Simulated behavior of the gaussian circuit. The current I_a determining the standard deviation of the gaussian is varied from 11nA to 31nA in steps of 2nA.

Table 6.2. Simulation results of the gaussian circuit

Circuit	*Gaussian*	
	Minimum	Maximum
Ref. I_a (nA)	11	31
Input current	$-10I_a$	$10I_a$
Power (μW)	18.14	24.36
Error (%)	8.78	13.15

6.2 Synthesis of dynamical systems

Let the dynamical system to be implemented be given by

$$\begin{aligned}\dot{\mathbf{x}}(t) &= f(\mathbf{x}(t), \mathbf{u}(t)) \\ \mathbf{y}(t) &= g(\mathbf{x}(t), \mathbf{u}(t))\end{aligned} \tag{6.6}$$

where $\mathbf{u}(t)$ is the input to the system, $\mathbf{x}(t)$ is the *state*, and $\mathbf{y}(t)$ is the output of the system. As discussed in Chapter 1, the existing methods proposed in [30, 16, 22, 31, 32, 33] all make use of integrators through the exponential state-space transformation.

The method proposed in this thesis is to use lowpass filter(s) for implementing dynamical systems. Here, the idea is to convert $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ into a set of low-pass filter– like equations of the form $\dot{\mathbf{x}} + D(\mathbf{x}, \mathbf{u})\mathbf{x} = \hat{f}(\mathbf{x}, \mathbf{u})$, where $D(\mathbf{x}, \mathbf{u})$ is a diagonal matrix whose diagonal elements may or may not depend on the state variable $\mathbf{x}$ and the inputs but is always constrained to be positive. This idea derives from the fact that a standard MITE low-pass filter shown in Figure 6.9 has a equation of the form

$$\frac{CU_{\mathrm{T}}}{\kappa}\dot{I}_y + I_{\tau 1}I_y = I_{\tau 2}I_x, \tag{6.7}$$

where it has been shown in Chapter 1 that the current $I_{\tau 1} = I_{\tau 1}(t)$ need not be constant for the equation to hold.

To implement Equation (6.7) for bidirectional input currents I_x, we use current splitting through a geometric current splitter. As before, we generate two positive currents I_{x+} and I_{x-} satisfying $I_x = I_{x+} - I_{x-}$ and $I_{x+}I_{x-} = I_b^2$. If we feed these currents through a lowpass filter, we get outputs I_{y+} and I_{y-} according to:

$$\begin{aligned}\frac{CU_{\mathrm{T}}}{\kappa}\dot{I}_{y+} + I_{\tau 1}I_{y+} &= I_{\tau 2}I_{x+} \\ \frac{CU_{\mathrm{T}}}{\kappa}\dot{I}_{y-} + I_{\tau 1}I_{y-} &= I_{\tau 2}I_{x-}\end{aligned} \tag{6.8}$$

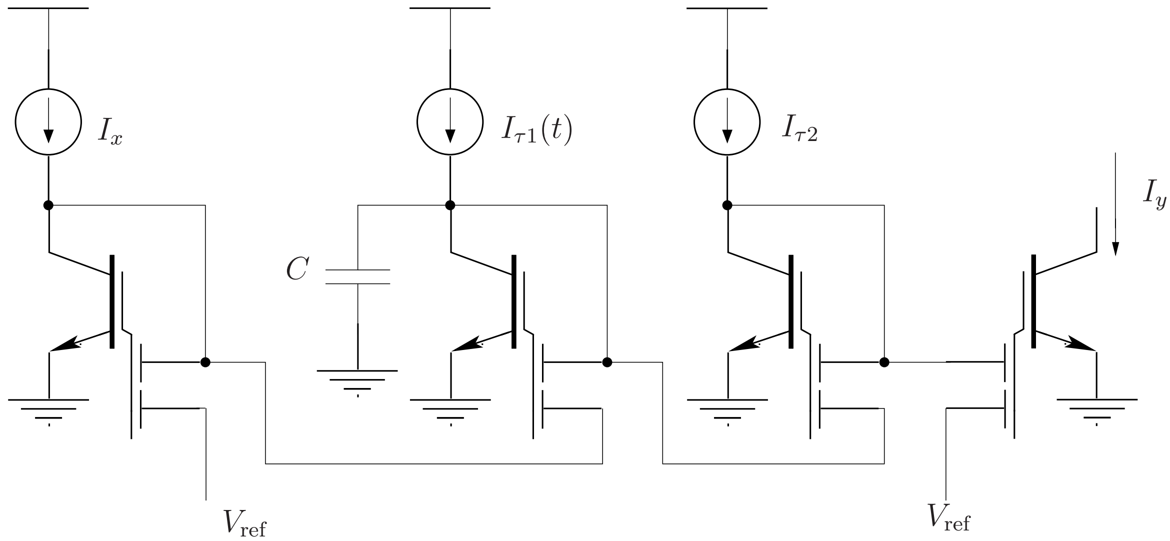


Figure 6.9. The standard MITE first-order lowpass filter. The filter obeys the equation $(CU_{\mathbf{T}})/\kappa \dot{I}_y + I_{\tau 1} I_y = I_{\tau 2} I_x$, where $I_{\tau 1}$ need not be constant.

It is clear that $I_y = I_{y+} - I_{y-}$ satisfies Equation (6.7). This bidirectional lowpass filter is shown in Figure 6.10. The consolidation done to remove a MITE in each single-ended lowpass filter should be noted. We will explore the proposed method in the following two systems:

6.2.1 The Lorenz system

The Lorenz attractor is given by the following set of first-order differential equations

$$\begin{aligned} \dot{x} &= \sigma(y - x) \\ \dot{y} &= x(\rho - z) - y \\ \dot{z} &= xy - \beta z \end{aligned} \tag{6.9}$$

It is easily seen that this set of equations can be converted into a set of lowpass filter equations:

$$\begin{aligned} \dot{x} + \sigma x &= \sigma y \\ \dot{y} + \ \ y &= x(\rho - z) \\ \dot{z} + \beta z &= xy \end{aligned} \tag{6.10}$$

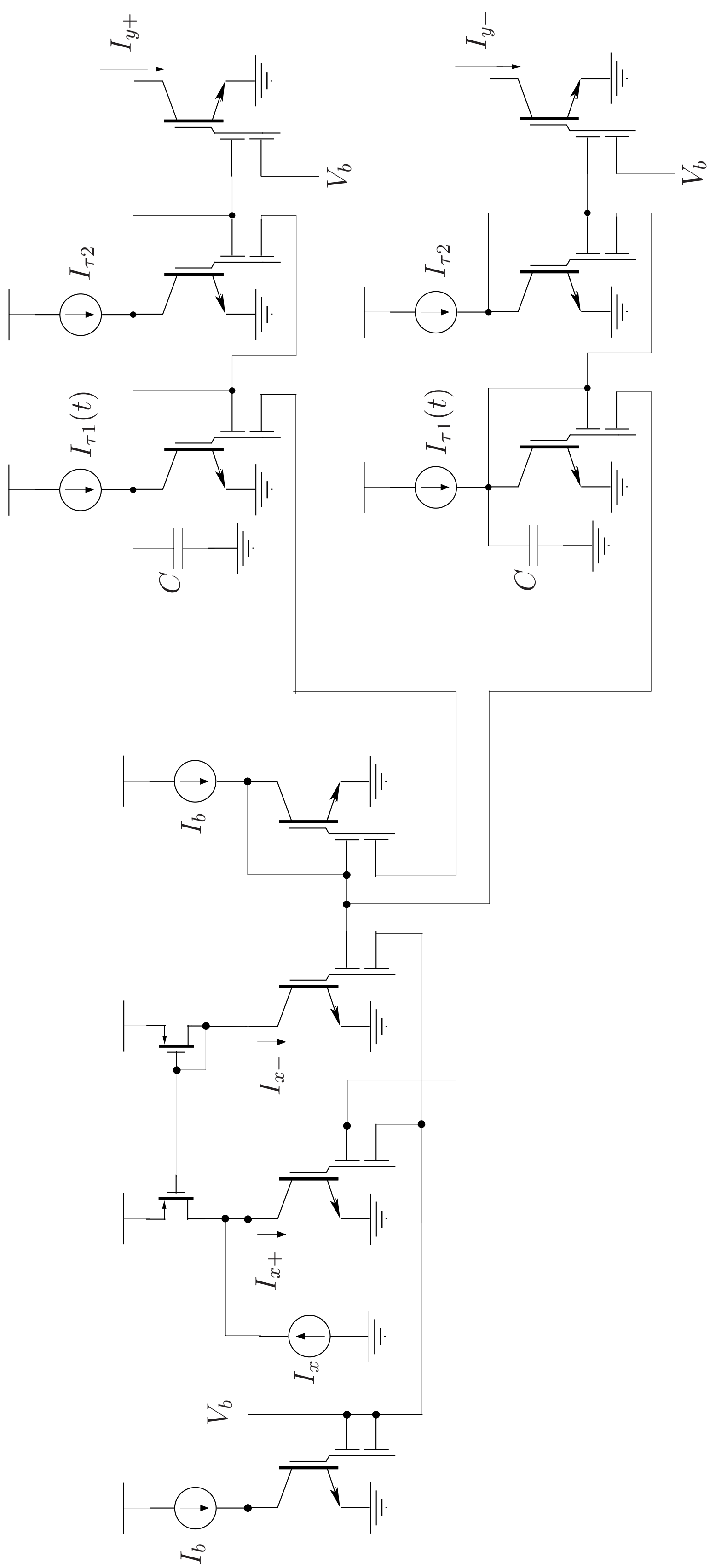


Figure 6.10. The bidirectional MITE first-order lowpass filter. The input I_x undergoes splitting and the positive current outputs are fed into firstorder lowpass filters. The filter obeys the equation $(CU_{\mathbf{T}})/\kappa \dot{I}_y + I_{\tau 1} I_y = I_{\tau 2} I_x$, where $I_y = I_{y+} - I_{y-}$. Here, $I_{\tau 1}$ need not be constant.

After replacing the dimensionless time t by t/τ and the signals by the ratios of currents to a scaling current I_a, we find the resultant lowpass filter equations to be

$$\begin{aligned}\frac{CU_\mathrm{T}}{\kappa}\dot{I}_x + (\sigma I_a)I_x &= (\sigma I_a)I_y \\ \frac{CU_\mathrm{T}}{\kappa}\dot{I}_y + \ (I_a)y &= (I_a)\frac{I_x(I_\rho - I_z)}{I_a} \\ \frac{CU_\mathrm{T}}{\kappa}\dot{I}_z + (\beta I_a)I_z &= (I_a)\frac{I_x I_y}{I_a},\end{aligned} \tag{6.11}$$

where C is chosen so that $\tau = CU_\mathrm{T}/(\kappa I_a)$. The nonlinearity in these equation is minimal: two products $I_x I_y$ and $I_x(I_\rho - I_z)$. Since the equations are in the form of bidirectional lowpass filters, the inputs $I_{\mathrm{in},x}$, $I_{\mathrm{in},y}$, and $I_{\mathrm{in},z}$ to the lowpass filters in Equation (6.11) are given by $I_{\mathrm{in},x} = I_y$, $I_{\mathrm{in},y} = I_x(I_\rho - I_z)/I_a$, and $I_{\mathrm{in},z} = I_x I_y/I_a$. This is shown in Figures 6.11 and 6.12. It is easily seen that $I_x = I_{x+} - I_{x-}$, $I_y = I_{y+} - I_{y-}$, and $I_z = I_{z+} - I_{z-}$ i.e., the state variables are given by the differences of the six variables that are the outputs of the lowpass filters. The currents $I_{\mathrm{in},x}$, $I_{\mathrm{in},y}$, and $I_{\mathrm{in},z}$ are implemented as

$$\begin{aligned}I_{\mathrm{in},x} &= I_{y+} - I_{y-} \\ I_{\mathrm{in},y} &= \frac{I_{x+}I_\rho}{I_a} + \frac{I_{x+}I_{z-}}{I_a} + \frac{I_{x-}I_{z+}}{I_a} - \frac{I_{x+}I_\rho}{I_a} - \frac{I_{x+}I_{z+}}{I_a} - \frac{I_{x-}I_{z-}}{I_a} \\ I_{\mathrm{in},z} &= \frac{I_{x+}I_{y+}}{I_a} + \frac{I_{x-}I_{y-}}{I_a} - \frac{I_{x+}I_{y-}}{I_a} - \frac{I_{x-}I_{y+}}{I_a}\end{aligned} \tag{6.12}$$

The products are implemented using the methods in the previous chapters and the corresponding circuit is shown in Figure 6.13. From the circuit simulation, the phase plot for $\sigma = 3$, $\beta = 1$, $\rho = 30$ is shown in Figure 6.14.

6.2.2 A sinusoidal oscillator with independent frequency and amplitude control

The sinusoidal oscillator dealt with here is a popular example in dynamical systems theory to illustrate the existence of limit cycles. The differential equation in (r, θ) coordinates is given by:

$$\begin{aligned}\dot{r} &= \mu r(\alpha^2 - r^2) \\ \dot{\theta} &= \omega_0\end{aligned} \tag{6.13}$$

which when converted into x, y coordinates transforms into

$$\begin{aligned}\dot{x} &= \mu x(\alpha^2 - r^2) - \omega_0 y \\ \dot{y} &= \mu y(\alpha^2 - r^2) + \omega_0 x\end{aligned} \tag{6.14}$$

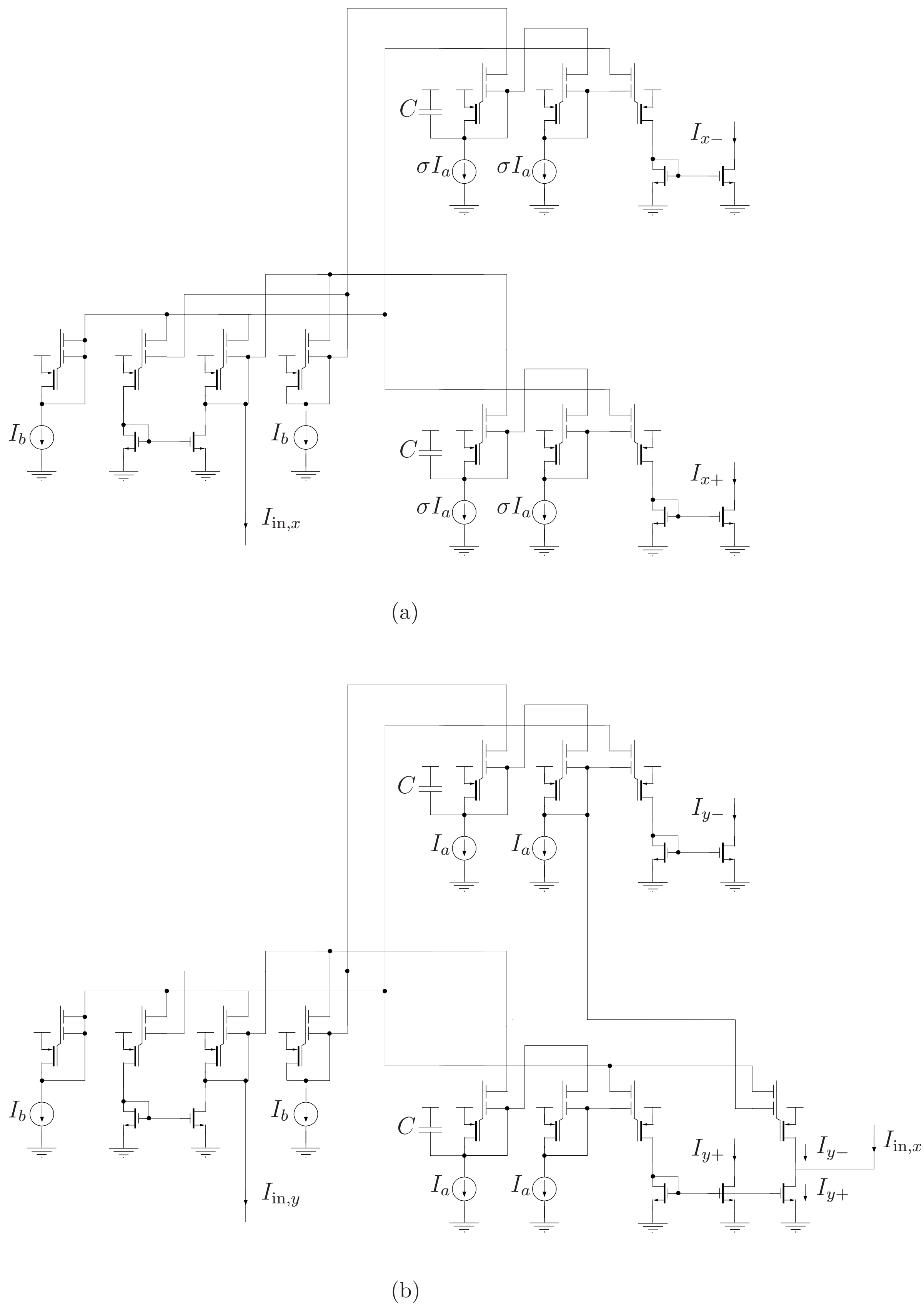
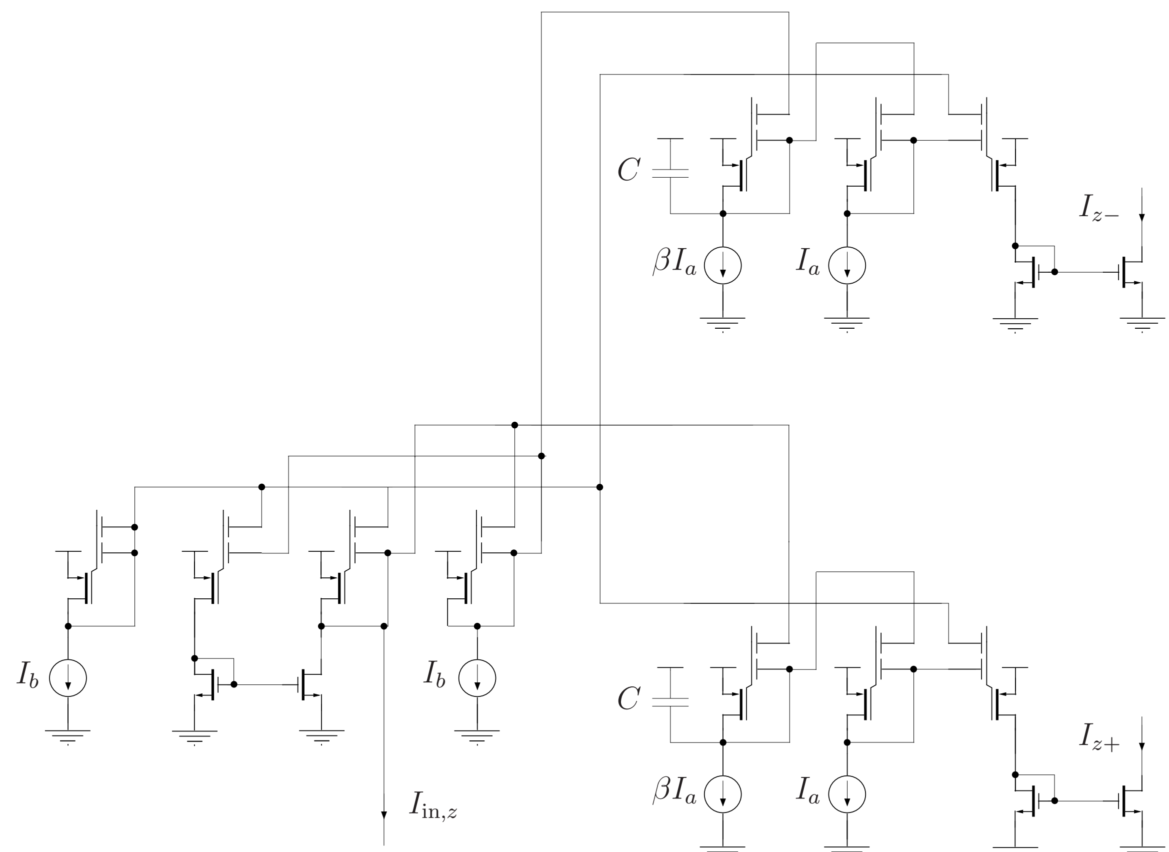
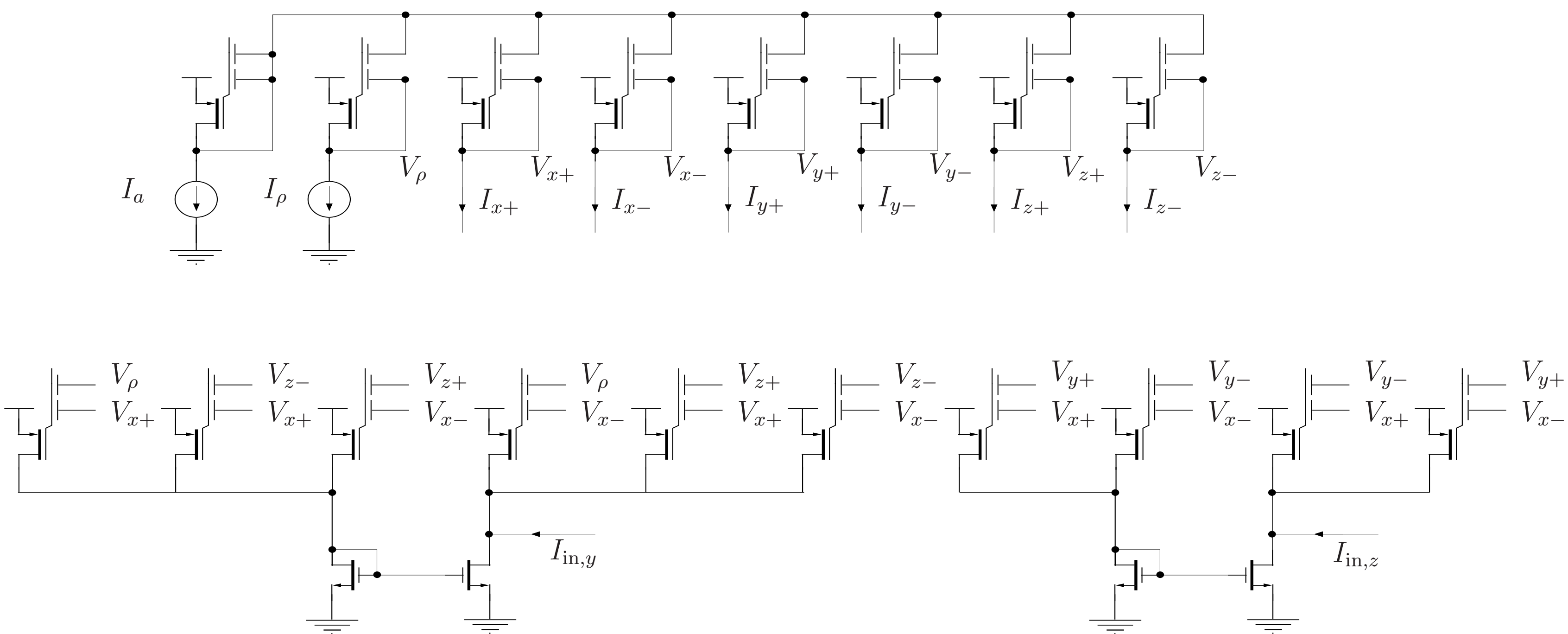


Figure 6.11. The part of the MITE circuit implementing the Lorenz system consisting of the bidirectional lowpass filters. (a) corresponds to the x equation and (b) corresponds to the y equation in Equation (6.11). The input $I_{\text{in},x}$ is also generated in (b).



(c)

Figure 6.12. The part of the MITE circuit implementing the Lorenz system consisting of the bidirectional lowpass filter corresponding to the z coordinate in Equation (6.11).



(d)

Figure 6.13. The static MITE circuit implementing the inputs $I_{\mathrm{in},y}$ and $I_{\mathrm{in},y}$, i.e., the right-hand side of the y and z equations in Equation (6.11).

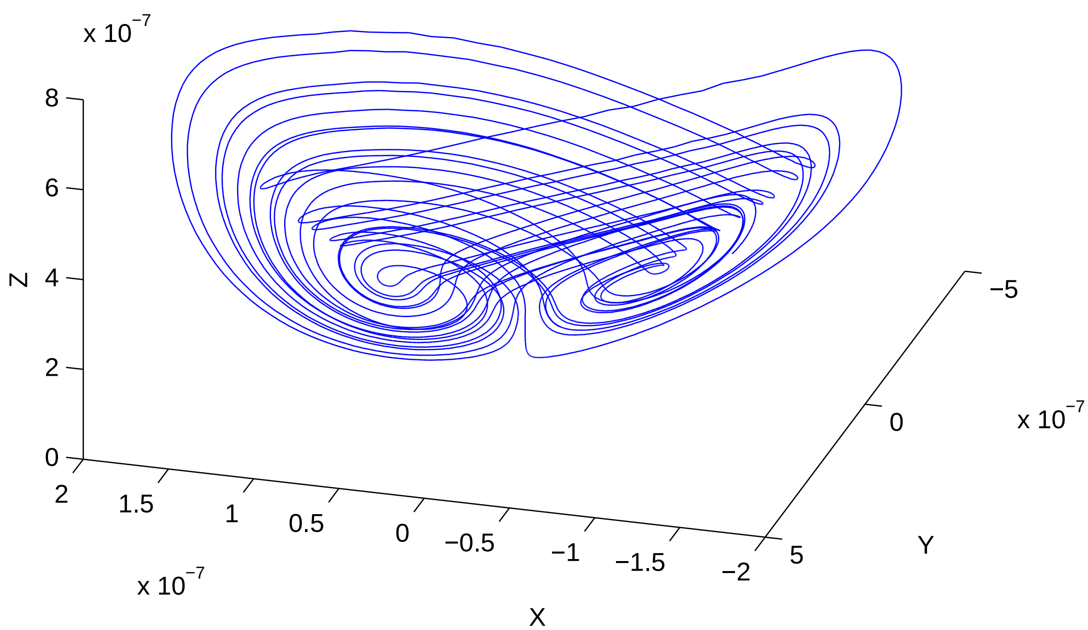


Figure 6.14. The results of simulation of the circuit of the Lorenz system for the parameter values $\sigma = 3$, $\beta = 1$, $\rho = 30$.

Consider the equivalent set of equations:

$$\begin{aligned}\dot{x} + \mu(\beta^2 + r^2)x &= \mu x(\alpha^2 + \beta^2) - \omega_0 y \\ \dot{y} + \mu(\beta^2 + r^2)y &= \mu y(\alpha^2 + \beta^2) + \omega_0 x\end{aligned} \tag{6.15}$$

This is a set of lowpass filters, one of the controlling currents of which is made dependent on the state. By adding β^2, we are ensuring that this term always remains positive, including at the origin. It should be noted that *this is not a quasi-static approximation* but follows from the derivation of the first-order lowpass filter itself.

Let us replace the signals by the ratios of currents to a scaling current I_b. It is clear that the form of Equation (6.15) is maintained if the currents I_x and I_y satisfy

$$\begin{aligned}\frac{CU_\mathrm{T}}{\kappa}\dot{I}_x + \left(\frac{I_\beta^2 + I_r^2}{I_b}\right) I_x &= I_a I_x - I_\omega I_y \\ \frac{CU_\mathrm{T}}{\kappa}\dot{I}_y + \left(\frac{I_\beta^2 + I_r^2}{I_b}\right) I_y &= I_a I_y + I_\omega I_x,\end{aligned} \tag{6.16}$$

where $I_r^2 = I_x^2 + I_y^2$. We need to choose an appropriate I_β so that $(I_r^2 + I_\beta^2)/I_b$ can be calculated easily. If we assume that I_x and I_y are passed through current splitters with geometric mean I_b, then we have positive currents I_{x+}, I_{x-}, I_{y+}, and I_{y-} satisfying $I_x = I_{x+} - I_{x-}$ and $I_y = I_{y+} - I_{y-}$ with $I_{x+}I_{x-} = I_{y+}I_{y-} = I_b^2$. Then $I_r^2 = I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2 - 4I_b^2$. If we choose $I_\beta = 2I_b$, then $I_r^2 + I_\beta^2 = I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2$, which is easily computed. Therefore, the lowpass filter form of the sinusoidal oscillator is

$$\begin{aligned}\frac{CU_\mathrm{T}}{\kappa}\dot{I}_x + \left(\frac{4I_b^2 + I_r^2}{I_b}\right) I_x &= I_a \left(I_x - \frac{I_\omega I_y}{I_a}\right) \\ \frac{CU_\mathrm{T}}{\kappa}\dot{I}_y + \left(\frac{4I_b^2 + I_r^2}{I_b}\right) I_y &= I_a \left(I_y + \frac{I_\omega I_x}{I_a}\right),\end{aligned} \tag{6.17}$$

Let us compute the amplitude and frequency of oscillation. Comparing Equation (6.17) and Equation (6.14), it is clear that the amplitude is found by equating the term multiplying I_x to 0 and the frequency is found by from the coefficient multiplying I_y. Therefore, $I_a = (I_\mathrm{amp}^2 + 4I_b^2)/I_b$ and hence the amplitude of oscillation is given by

$$I_\mathrm{amp} = \sqrt{I_a I_b - 4I_b^2}$$

Similarly, the frequency of oscillation ω_0 is given by

$$\omega_0 = \frac{I_\omega \kappa}{CU_\mathrm{T}}$$

6.2.2.1 *Implementation details*

The input currents to the bidirectional lowpass filters are given by $I_{\text{in,x}} = I_x - I_\omega I_y/I_a$ and $I_{\text{in,y}} = I_y + I_\omega I_x/I_a$. Though the differences of the outputs of the lowpass filters are I_x and I_y, the outputs need have a constant geometric mean. Since we have assumed the presence of I_{x+}, I_{x-}, I_{y+} satisfying a geometric mean constraint, two separate geometric current splitters are required with inputs I_x and I_y. In terms of the positive currents, we have

$$\begin{aligned} I_{\text{in,x}} &= I_{x+} + \frac{I_\omega I_{y-}}{I_a} - I_{x-} - \frac{I_\omega I_{y+}}{I_a} \\ I_{\text{in,y}} &= I_{y+} + \frac{I_\omega I_{x+}}{I_a} - I_{y-} - \frac{I_\omega I_{x-}}{I_a} \end{aligned} \tag{6.18}$$

Further, we also require I_{x+}^2/I_b, I_{x-}^2/I_b, I_{y+}^2/I_b, and I_{y-}^2/I_b for computing $(I_r^2 + 4I_b^2)/I_b$. Along with the geometric current splitter constraints, we need a MITE network implementing the following equations:

$$\begin{aligned} &I_{x-}I_{x+} = I_b I_b; \quad I_{y-}I_{y+} = I_b I_b; \\ &I_{o1} = \frac{I_{x+}^2}{I_b}; \quad I'_{o1} = \frac{I_{y+}^2}{I_b} \\ &I_{o2} = \frac{I_{x-}^2}{I_b}; \quad I'_{o2} = \frac{I_{y-}^2}{I_b} \\ &I_{o3} = \frac{I_\omega I_{x+}}{I_a}; \quad I'_{o3} = \frac{I_\omega I_{y+}}{I_a} \\ &I_{o4} = \frac{I_\omega I_{x-}}{I_a}; \quad I'_{o4} = \frac{I_\omega I_{y-}}{I_a} \end{aligned} \tag{6.19}$$

Since the constraints involving x and y are exactly the same except for the substitution of one for another, it is enough to find a POPL network solving the equations involving x. The reason why we write I_b^2 as the product of two currents is because we are interested in a 2-MITE implementation, for otherwise the optimal synthesis procedure of Chapter 3 gives a fan-in of 4 as the minimum fan-in required to implement all the equations together. Taking the currents in the sequence I_b, I_b, I_ω, I_a, I_{x+}, I_{x-}, I_{o1}, I_{o2}, I_{o3}, and I_{o4}, we find the translinear loop matrix A and the *only* solution connectivity matrix Z to be:

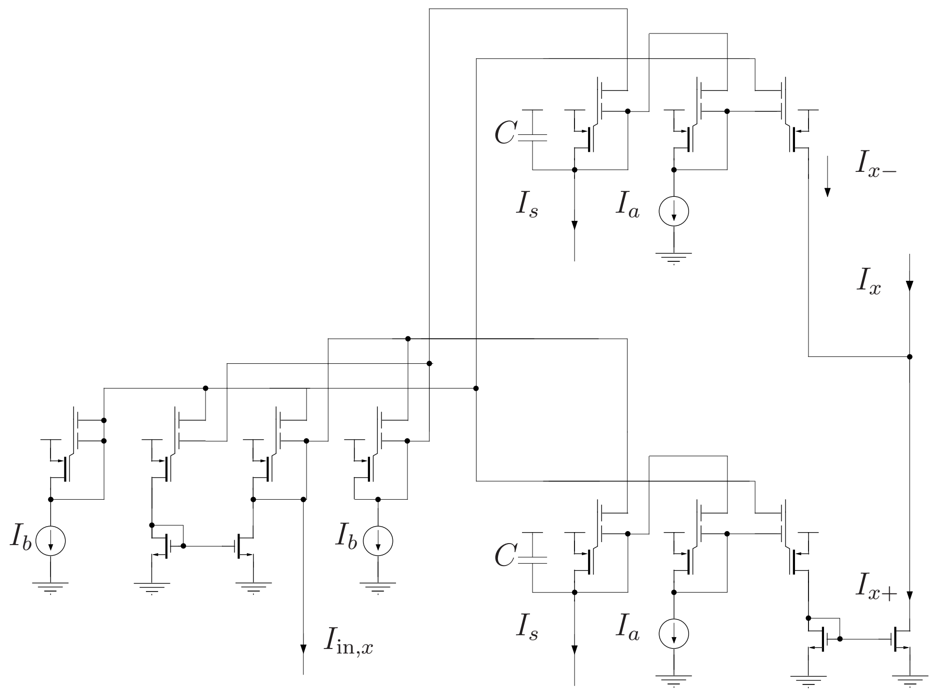
$$A = \begin{bmatrix} 1 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 2 & 0 & -1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 2 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 1 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$Z = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 \\ 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

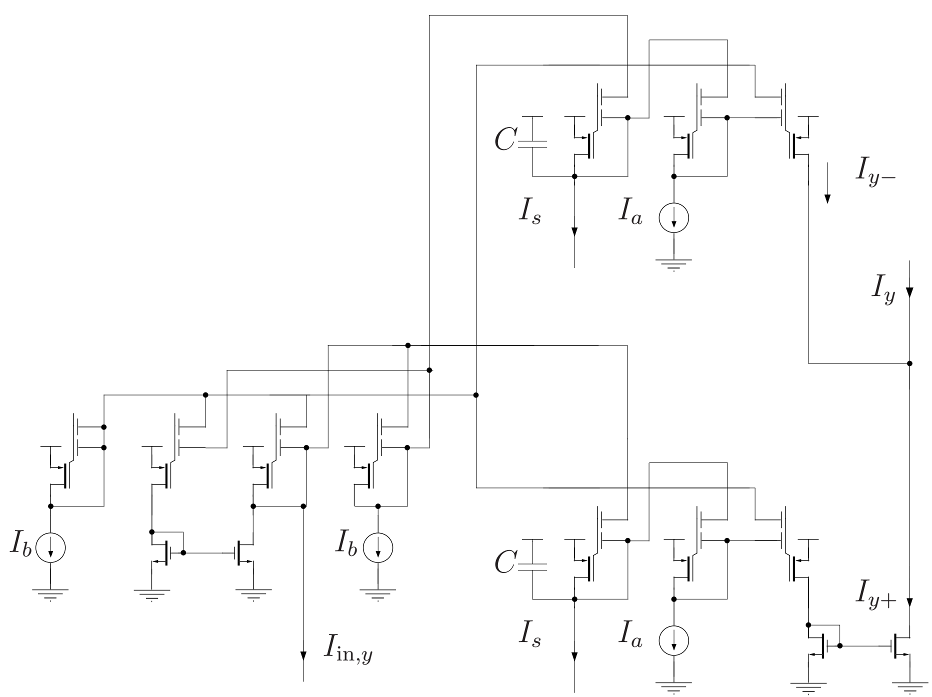
The circuit of the lowpass filters is shown in Figure 6.15 and the POPL network implementing the above relations is shown in Figure 6.16. The simulations results for varying amplitudes is shown in Figure 6.17. The current I_b is varied from 5nA to 25nA in steps of 2nA while $I_a = 8I_b$ and $I_\omega = 20$nA. Hence the amplitude should increase as $I_{\text{amp}} = \sqrt{(8I_b^2 - 4I_b^2)} = 2I_b$. The deviation of the phase plot from the ideal is also clearly observed.

6.2.3 Chip fabrication

The arctan, gaussian, Lorenz, and the sinusoidal oscillator blocks were implemented in 0.5μ technology. The layout of the chip is shown in Figure 6.18. As shown in the previous sections, the simulations of the blocks show that they are functional. However, the fabricated chip itself had problems unrelated to the working of each block. Programming the floating-gate MOSFETs in the chip was found to be not possible mainly because of latch-up issues. Hence, we could neither prove nor disprove that the synthesized blocks in the chip were functional. However, the author believes that this does not affect the importance or the contributions of this thesis which is in the mathematically sound and systematic methods developed for MITE network synthesis.

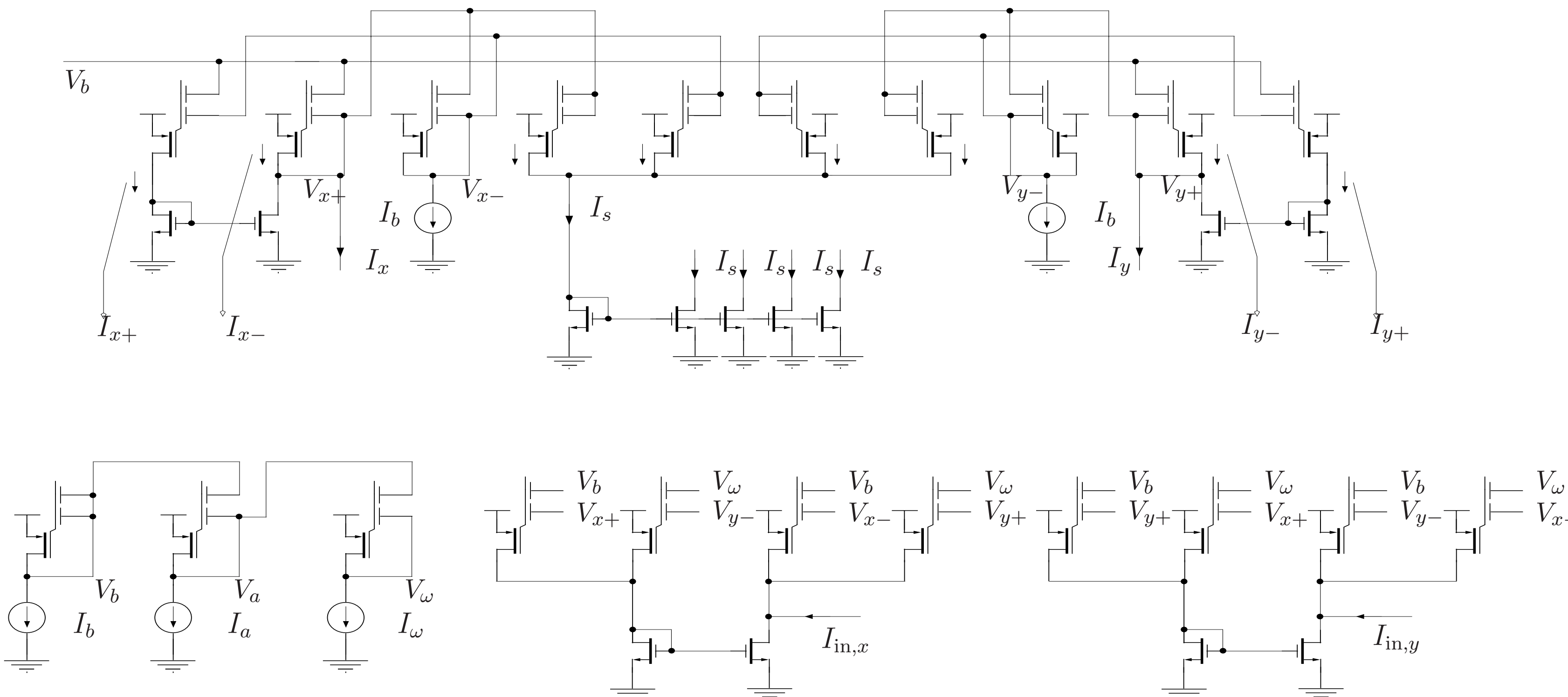


(a)



(b)

Figure 6.15. The part of the MITE circuit implementing the sinusoidal oscillator consisting of the bidirectional lowpass filters. (a) corresponds to the x equation and (b) corresponds to the y equation in Equation (6.17). **Here the time varying current** $I_s = (I_r^2 + I_\beta^2)/I_b = (I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2)/I_b$



(c)

Figure 6.16. The part of the MITE circuit implementing the sinusoidal oscillator consisting of POPL networks implementing $I_{\mathrm{in},x}$, $I_{\mathrm{in},y}$, and $I_s = (I_r^2 + I_\beta^2)/I_b = (I_{x+}^2 + I_{x-}^2 + I_{y+}^2 + I_{y-}^2)/I_b$

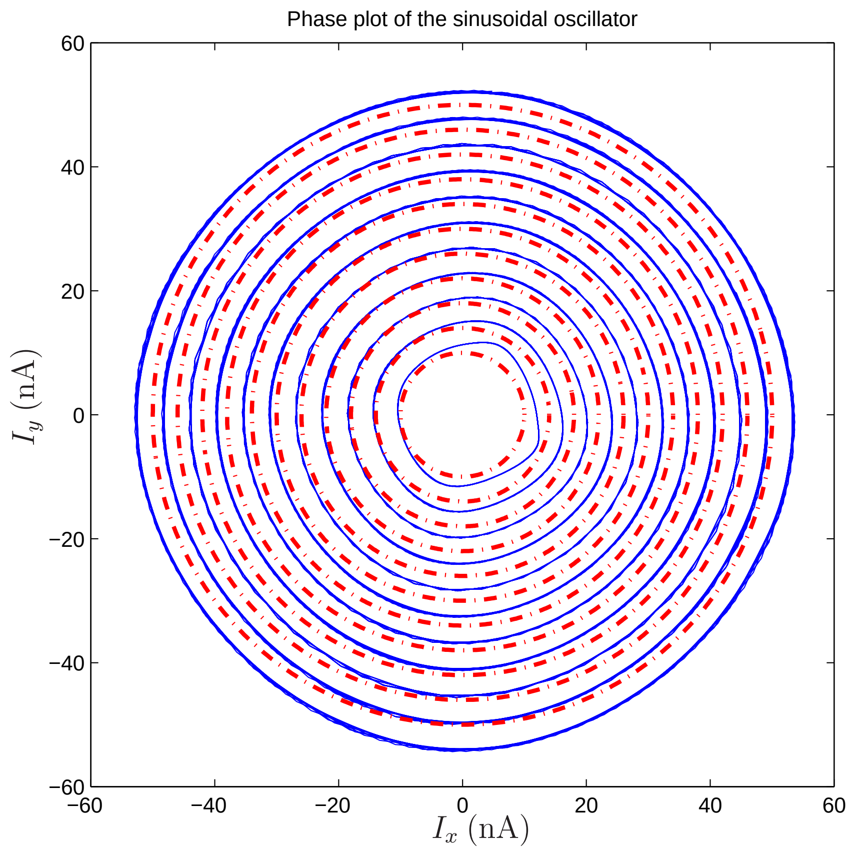


Figure 6.17. The results of circuit simulation of the sinusoidal oscillator for varying amplitudes. The current I_b is varied from 5nA to 25nA in steps of 2nA while $I_a = 8I_b$ and $I_\omega = 20$nA. The ideal behaviour is given in dotted lines for comparison.

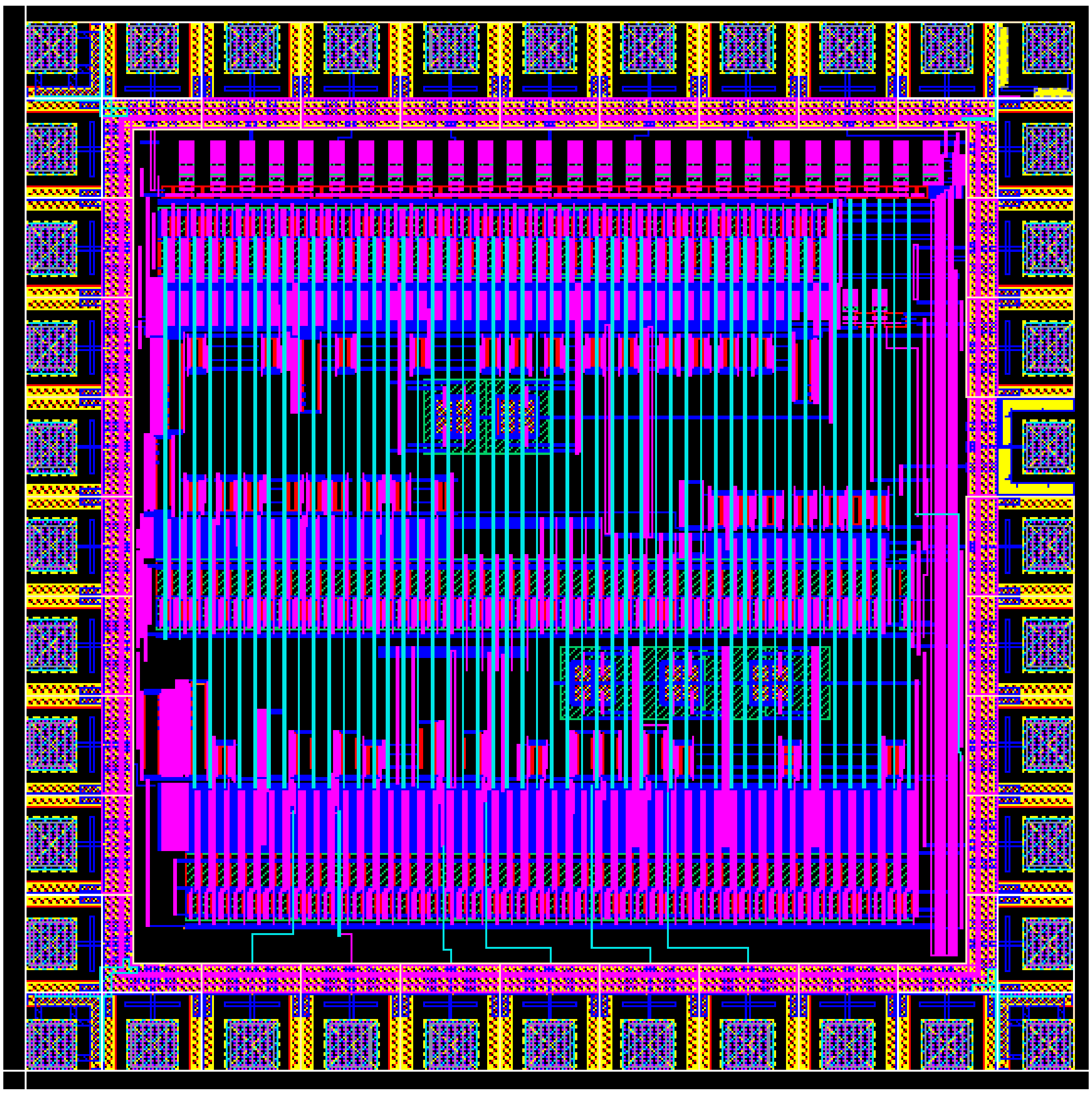


Figure 6.18. The layout of the chip in 0.5μ technology containing the arctan, gaussian, Lorenz, and the sinusoidal oscillator blocks

CHAPTER 7
CONCLUSIONS AND FUTURE RESEARCH

The main goal of this research is the automated and optimal synthesis of multiple-input translinear element circuits. From a circuit-theoretic point of view, this thesis is to be viewed as a step towards mapping the set of algebraic functions and differentially algebraic equations to the class of circuits with only the following components:

1. n-input MITEs

2. current mirrors

3. capacitors

While some methods to find a circuit belonging to the above class corresponding to a given function have already been developed, the novel contribution of this thesis is the development of systematic synthesis procedures to find those circuits that, in addition to implementing the function, also *optimize* it in some sense.

Since the aim is not to design a *single* circuit, but to design a class of circuits satisfying a general relationship (like the product–of–power law (POPL)), a detailed mathematical treatment is inevitable. The path chosen for this research is to proceed from ideal assumptions about the MITE and then to add the nonidealities in the order of significance.

7.1 Contributions of this research

1. Derived condition dependent only upon the topology of a POPL MITE network that ensures that the operating point is unique. The effect of floating-gate capacitor mismatch is taken into account so that it does not affect the uniqueness of the operating point.

2. Derived an improved condition for stability of a POPL network. This condition is dependent only upon the topology of the MITE network and on the ratio C_p/C of the parasitic capacitance seen by the floating-gate to the floating-gate capacitance.

3. Developed a systematic synthesis procedure that generates POPL networks implementing a given system of translinear loop equations. The procedure is optimal in

the sense that the generated MITE networks utilize the minimum possible number of MITEs and further have minimum fan-in amongst those networks that implement the translinear loop equations using the minimum possible number of MITEs.

4. Characterized 2-MITE POPL networks in terms of their Coates graphs. Developed a procedure to obtain the power matrix of a 2-MITE POPL network from its Coates graph by observation alone.

5. Showed that under mild conditions, a 2-MITE POPL network automatically satisfies both the uniqueness criterion and D-stability criterion even under small pertubations of the floating-gate capacitor values.

6. Developed necessary conditions that a power matrix must satisfy if it can be implemented by a 2-MITE POPL network.

7. Developed necessary and sufficient conditions that completely characterize single-output 2-MITE POPL networks.

8. Developed a synthesis procedure to implement any given single-output POPL function using a 2-MITE POPL network using the minimum required number of copies of the input currents.

9. For use in a MITE FPAA, developed a single-output 2-MITE POPL "basic structure" with n inputs that can implement most 2-MITEable POPL functions that have at most n inputs by changing only the input-gate connections of the output MITE.

10. Developed the concept of the modified Coates graph representation of a 2-MITE POPL network that can be used to find the general Coates graphs that can represent 2-MITE POPL networks with a fixed number of outputs. This is shown to be useful in characterizing 2-MITE POPL networks with two outputs, for example.

11. Developed conditions and a procedure to synthesize log-domain filters that necessarily avoids multiple operating points.

12. Developed a procedure to synthesize, using MITEs, static functions with bidirectional inputs as a function of one positive current variable. To illustrate this, the synthesis of the arctangent and the gaussian function are described.

13. Developed a new method to synthesize dynamic functions by the use of first-order lowpass filters. Exemplary syntheses include that of a Lorenz system and a sinusoidal oscillator with independent amplitude and frequency control.

7.2 Future Research

Future research in synthesis of MITE networks can take a theoretical as well as a practical form.

7.2.1 Future Theoretical research

1. While the synthesis of 2-MITE single-output POPL networks is complete, the synthesis of multiple-output 2-MITE POPL networks seems complex but should also be highly interesting as well as useful in 2-MITE synthesis. For example, it is easy to check if a given multiple-output translinear loop matrix A is 2-MITEable or not using the method of diophantine equations in Chapter 3. However, for synthesis purposes, it is best if one has a compact condition like Theorem 4.5.1 for the single-output case. This problem itself can be described as follows:

   Given $A \in \mathcal{M}_{l,m}(\mathbb{Z})$, does there exist a $Z \in \mathcal{M}_{m,n}(\mathbb{N})$ such that

   (a) $AZ = 0$

   (b) $Z\mathbf{1}_n = 2\mathbf{1}_m$

   (c) $Z_{ii} > 0$ for $i \in [1 : n]$

   Once this characterization is complete, when A is not 2-MITEable, the optimal number of copies of the input currents should be found so that the resultant translinear loop matrix is 2-MITEable.

2. The presence of the floating-gate capacitors can affect the frequency response of the MITE system under consideration. Finding those MITE structures that minimize

the effect of these and other parasitics should improve the frequencies of operation of MITE circuits considerably. A good starting point would be to deal with these capacitors in POPL networks.

3. Characterizing those functions that are implementable using static and dynamic MITE networks should aid considerably in developing synthesis approaches.

4. Some metric(s) that can be used to compare MITE networks, and especially POPL networks, with respect to noise, bandwidth, and sensitivity to temperature can be developed.

7.2.2 Future Practical Research

1. Developing an automated on-chip floating-gate programming method that is also minimal in chip area.

2. Utilizing the methods described here to completely automate the equation-to-layout process for *any* algebraic function, not just POPL functions. Similar programs for dynamic systems should also be useful. This should also find application in reconfigurable systems, i.e., the MITE FPAA.

CHAPTER 8

NOTATION

1. $\mathbb{C}, \mathbb{R}, \mathbb{Q}, \mathbb{Z},$ and $\mathbb{N}$ denote the set of complex numbers, reals, rationals, integers, and nonnegative integers, respectively.

2. The set of all $m \times n$ matrices whose elements are restricted to $\mathbb{F} \subseteq \mathbb{C}$ is denoted by $\mathcal{M}_{m,n}(\mathbb{F})$.

3. $\mathcal{M}_{n,n}(\mathbb{F})$ and $\mathcal{M}_{n,1}(\mathbb{F})$ are abbreviated to $\mathcal{M}_n(\mathbb{F})$ and $\mathbb{F}^n$, respectively.

4. $A > B$ ($A \geq B$) means that the elements of $A - B$ are positive (nonnegative).

5. $A \gg B$ means that $A \geq B$ and that there exist elements A_{ij} and B_{ij} such that $A_{ij} > B_{ij}$.

6. If $m \leq n$, $[m : n]$ is the set $\{m, m+1, \ldots, n\}$.

7. If $A \in \mathcal{M}_{m,n}(\mathbb{F})$, $\alpha \subseteq [1 : m]$, and $\beta \subseteq [1 : n]$, then $A(\alpha, \beta)$ is the matrix formed by the rows and columns of A indexed by α and β, respectively.

8. The phrase *diagonal matrix* $D > 0$ (*diagonal matrix* $D \geq 0$) means that the matrix D is a diagonal matrix with only positive (nonnegative) entries along the diagonal.

9. I_n is the $n \times n$ identity matrix.

10. $\mathbf{1}_n$ denotes the $n \times 1$ vector with all elements being 1.

11. if $f\colon \mathbb{A} \to \mathbb{B}$, and if $\mathbf{x} = [x_i] \in \mathbb{A}^n$, then $f(\mathbf{x})$ denotes the vector $[f(x_i)] \in \mathbb{B}^n$.

12. The standard determinant expansion of a matrix $A = [a_{ij}] \in \mathcal{M}_n(\mathbb{F})$ is given by $\det(A) = \sum_\sigma \operatorname{sign}\sigma \prod_{i=1}^n a_{i\sigma(i)}$ where the sum runs over all $n!$ permutations σ of $[1 : n]$.

13. a function $f(x_1, x_2, \ldots, x_n)$ is *multilinear(multiaffine)* is said if it is linear(affine) in each variable when the other variables are kept constant.

14. If v is a row or column vector of order n, then $\operatorname{diag}(v)$ is the $n \times n$ diagonal matrix with v in the diagonal.

REFERENCES

[1] B. Gilbert, "Translinear circuits: a proposed classification," *Electronics Letters*, vol. 11, pp. 14–16, 1975.

[2] ——, "Translinear circuits: An historical review," *Analog Integrated Circuits and Signal Processing*, vol. 9, no. 2, pp. 95–118, Mar. 1996.

[3] J. Mulder, W. A. Serdijn, A. C. van der Woerd, and A. H. M. van Roermund, *Dynamic Translinear and Log-Domain Circuits: Analysis and Synthesis*. Boston: Kluwer, 1999.

[4] E. Seevinck, *Analysis and Synthesis of Translinear Integrated Circuits*. Amsterdam: Elsevier, 1988.

[5] B. Gilbert, "A precise four-quadrant multiplier with subnanosecond response," *IEEE Journal of Solid-State Circuits*, vol. 3, pp. 365–373, Dec. 1968.

[6] S. Ashok, "Integrable sinusoidal frequency doubler," *IEEE Journal of Solid-State Circuits*, vol. SC-11, no. 2, pp. 341–343, 1976.

[7] M. T. Abuelma'atti and S. M. Abed, "A translinear circuit for analogue function synthesis based on a taylor series," *International Journal of Electronics*, vol. 86, pp. 1341–1348, 1999.

[8] D. R. Frey, "Exponential state space filters: A generic current mode-design strategy," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 43, pp. 34–42, Jan. 1996.

[9] ——, "State-space synthesis and analysis of log-domain filters," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 45, pp. 1205–1211, 1998.

[10] J. Mulder, A. C. V. der Woerd, W. A. Serdijn, and A. H. M. V. Roermund, "An rms-dc converter based on the dynamic translinear principle," *IEEE Journal of Solid-State Circuits*, vol. 32, pp. 1146–1150, July 1997.

[11] B. A. Minch, C. Diorio, P. Hasler, and C. A. Mead, "Translinear circuits using subthreshold floating-gate MOS transistors," *Analog Integrated Circuits and Signal Processing*, vol. 9, no. 2, pp. 167–179, 1996.

[12] B. A. Minch, "Analysis, synthesis, and implementation of networks of multiple-input translinear elements," Ph.D. dissertation, California Institute of Technology, May 1997.

[13] B. A. Minch, P. Hasler, and C. Diorio, "The multiple-input translinear element: A versatile circuit element," in *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, Monterey, CA, May 1998, pp. 527–530.

[14] ——, "Synthesis of multiple-input translinear element networks," in *Proceedings of the International Symposium on Circuits and Systems*, vol. 2, Orlando, FL, May 1999, pp. 236–239.

[15] B. A. Minch, "Floating-gate log-domain filter," in *Proceedings of the International Symposium on Circuits and Systems*, vol. II, Orlando, 1999.

[16] ——, "Multiple-input translinear element log-domain filters," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 29–36, Jan. 2001.

[17] B. A. Minch, P. Hasler, and C. Diorio, "Multiple-input translinear element networks," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 48, no. 1, pp. 20–28, Jan. 2001.

[18] B. A. Minch, "Synthesis of static and dynamic multiple-input translinear element networks," *IEEE Transactions on Circuits and Systems*, vol. 51, no. 2, pp. 409–421, Feb. 2004.

[19] ——, "Construction and transformation of multiple-input translinear element networks," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 50, pp. 1530–1537, Dec. 2003.

[20] ——, "Synthesis of dynamic multiple-input translinear element networks," in *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, geneva, Switzerland, May 2000, pp. 483–486.

[21] ——, "Analysis and synthesis of static translinear circuits," Computer Systems Laboratory, Cornell University, Ithaca, NY, Technical Report CSL-TR-2000-1002, 2000.

[22] ——, "Synthesis of static and dynamic multiple-input translinear element networks," Computer Systems Laboratory, Cornell University, Ithaca, NY, Technical Report CSL-TR-2002-1024, 2002.

[23] H. C. Nauta, "An integrated gamma corrector," *IEEE Journal of Solid-State Circuits*, vol. SC-16, no. 3, pp. 238–241, 1981.

[24] X. Arreguit, E. A. Vittoz, and M. Merz, "Precision compressor gain controller in cmos technology," *IEEE Journal of Solid-State Circuits*, vol. SC-22, no. 3, pp. 442–445, 1987.

[25] H.-J. Lo, G. Serrano, P. Hasler, D. V. Anderson, and B. A. Minch, "Programmable multiple input translinear elements," in *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, May 2004, pp. I–757 – I–760.

[26] K. M. Odame, E. J. McDonald, and B. A. Minch, "Highly linear, wide-dynamic-range multiple-input translinear element networks," in *Proc. of the Thirty-Seventh Annual Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2003, pp. 2036 – 2040.

[27] L. O. Chua, C. A. Desoer, and E. S. Kuh, *Linear and Nonlinear Circuits*. New York: McGraw-Hill Book Company, 1987.

[28] R. A. Horn and C. R. Johnson, *Matrix Analysis*. Cambridge, UK: Cambridge University Press, 1985.

[29] B. A. Minch, private communication, 2006.

[30] ——, “Synthesis of multiple-input translinear element log-domain filters,” in *Proceedings of the International Symposium on Circuits and Systems*, vol. II, Orlando, FL, 1999, pp. 697–700.

[31] E. J. McDonald and B. A. Minch, “Multi-level simulation of a translinear analog adaptive filter,” in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 4, May 2002, pp. IV–3992 – IV–3995.

[32] ——, “Synthesis of a translinear analog adaptive filter,” in *Proceedings of the International Symposium on Circuits and Systems*, vol. 3, May 2002, pp. III–321 – I–324.

[33] ——, “Synthesis of translinear analog signal processing systems,” in *Proc. 45th Midwest Symposium on Circuits and Systems, MWSCAS 2002*, vol. 1, Aug. 2002, pp. I–204 – I–207.

[34] K. M. Odame and B. A. Minch, “The translinear principle: a general framework for implementing chaotic oscillators,” *International Journal of Bifurcation & Chaos*, vol. 15, pp. 2559–2568, 2005.

[35] A. N. Willson, Jr, *Nonlinear Networks: Theory and Analysis*. New York: IEEE Press, 1975.

[36] J. L. Gouzé, “Positive and negative circuits in dynamical systems,” *Journal of Biological Systems*, vol. 6, no. 1, pp. 11–15, 1998.

[37] J. M. Ortega and W. C. Rheinboldt, *Iterative solution of nonlinear equations in several variables*. New York: Academic Press, 1970.

[38] A. N. Willson, Jr, “New theorems on the equations of nonlinear DC transistor networks,” *Bell System Technical Journal*, vol. 49, pp. 1713–1738, Oct. 1970.

[39] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*. Cambridge, UK: Cambridge University Press, 1991.

[40] I. W. Sandberg and A. N. Willson, Jr, “Some theorems on properties of dc equations of nonlinear networks,” *Bell System Technical Journal*, vol. 48, pp. 1–34, Jan. 1969.

[41] I. W. Sandberg, “Global inverse function theorems,” *IEEE Transactions on Circuits and Systems*, vol. 27, no. 11, pp. 998–1004, Nov. 1980.

[42] I. W. Sandberg and A. N. Willson, Jr, “Some network-theoretic properties of nonlinear dc transistor networks,” *Bell System Technical Journal*, vol. 48, pp. 1293–1311, May 1969.

[43] L. Kronenberg, W. Mathis, and L. Trajkovic, “Method PFBS for the analysis of transistor circuits: proof of uniqueness,” in *Proceedings of the Mixed Design of Integrated Circuits and Systems*, Gdynia, Poland, June 2000, pp. 145–148.

[44] L. Kronenberg, L. Trajkovic, and W. Mathis, “Analysis of feedback structures in FET circuits,” in *X International Symposium on Theoretical Electrical Engineering*, Magdeburg, Germany, Sept. 1999, pp. 445–450.

[45] ——, “Analysis of feedback structures and their effect on multiple dc operating points,” in *European Circuit Theory and Design Conference*, Stresa, Italy, Aug. 1999, pp. 683–686.

[46] A. Reibiger, W. Mathis, T. Nahring, L. Kronenberg, and L. Trajkovic, “Mathematical foundations of the TC-method for computing multiple dc-operating points,” in *Proc. XI. International Symposium on Theoretical Engineering*, Linz, Austria, Aug. 2001.

[47] C. R. Johnson, “Sufficient conditions for D-stability,” *Journal of Economic Theory*, vol. 9, pp. 53–62, 1974.

[48] B. E. Cain, “Real 3×3 D-stable matrices,” *Journal of Research of the National Bureau of Standards-B. Mathematical Sciences*, vol. 80-B, pp. 75–77, 1976.

[49] E. Kaszkurewicz and A. Bhaya, *Matrix diagonal stability in systems and computation.* Boston: Birkhäuser, 2000.

[50] L. O. Chua and P. Y. Lin, *Computer-Aided Analysis of Electronic Circuits: Algorithms and Computational Techniques.* Prentice Hall Professional Technical Reference, 1975.

[51] S. Subramanian and P. Hasler, “Uniqueness of the operating point in MITE circuits,” in *Proc. of the Thirty-Eighth Annual Asilomar Conference on Signals, Systems and Computers*, vol. 2, Nov. 2004, pp. 2218–2222.

[52] M. Tsatsomeros and L. Li, “A recursive test for P-matrices,” *BIT Numerical Mathematics*, vol. 40, no. 2, pp. 410–414, June 2000.

[53] S. Subramanian, D. V. Anderson, and P. Hasler, “Synthesis of static multiple input multiple output MITE networks,” in *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, May 2004, pp. I–189 – I–192.

[54] A. P. Tomás, “On solving linear diophantine constraints,” Ph.D. dissertation, Universidade do Porto, 1997.

[55] E. Contejean and H. Devie, “An efficient algorithm for solving systems of diophantine equations,” *Information and Computation*, vol. 113, no. 1, pp. 143–172, August 1994.

[56] E. Contejean, “Solving linear diophantine constraints incrementally,” in *Proc. of the Tenth Int. Conf. on Logic Programming*, ser. Logic Programming, D. S. Warren, Ed. Budapest, Hungary: MIT Press, June 1993, pp. 532–549.

[57] S. Subramanian, D. V. Anderson, P. Hasler, and B. A. Minch, “Optimal synthesis of MITE translinear loops,” in *Proceedings of the International Symposium on Circuits and Systems*, May 2007.

[58] W.-K. Chen, *Graph theory and its engineering applications.* Singapore: World Science Pub., 1997.

[59] D. N. Abramson, J. D. Gray, S. Subramanian, and P. Hasler, “A field-programmable analog array using translinear elements,” in *Proc. of the Fifth International Workshop on System-on-Chip for Real-Time Applications*, July 2005, pp. 425–428.

[60] J. D. Gray, “Application of floating-gate transistors in field programmable analog arrays,” Master’s thesis, Georgia Institute of Technology, 2005.

[61] D. D. N. Abramson, "MITE architectures for reconfigurable analog arrays," Master's thesis, Georgia Institute of Technology, 2004.

[62] A. Berman and D. Hershkowitz, "Matrix diagonal stability and its implications," *SIAM Journal on Algebraic and Discrete Methods*, vol. 4, no. 3, pp. 377–382, 1983.

[63] D. Hershkowitz, "Recent directions in matrix stability," *Linear Algebra and its Applications*, vol. 171, pp. 161–186, July 1992.

[64] G. P. Barker, A. Berman, and R. J. Plemmons, "Positive diagonal solutions to the Lyapunov equations," *Linear and Multilinear Algebra*, vol. 5, no. 3, pp. 249–256, 1978.

[65] M. J. Tsatsomeros and L. Li, "A Recursive Test for P-Matrices," *BIT Numerical Mathematics*, vol. 40, no. 2, pp. 404–408, 2000.

[66] R. M. Fox and M. Nagarajan, "Multiple operating points in a CMOS log-domain filter," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 6, pp. 705–710, 1999.

[67] J. Tow, "Design and evaluation of shifted-companion-form active filters," *Bell System Technical Journal*, vol. 54, no. 3, pp. 545–568, 1975.

[68] C. T. Fike, *Computer Evaluation of Mathematical Functions*, ser. Series in Automatic Computation. New Jersey: Prentice Hall, Inc., 1968.

[69] E. Seevinck, W. D. Jager, and P. Buitendijk, "A low-distortion output stage with improved stability for monolithic power amplifiers," *Solid-State Circuits, IEEE Journal of*, vol. 23, no. 3, pp. 794–801, 1988.

[70] E. Seevinck, "Companding current-mode integrator: A new circuit principle for continuous-time monolithic filters." *Electronics Letters*, vol. 26, no. 24, pp. 2046–2048, 1990.

[71] D. Frey, "Current mode class AB second order filter," *Electronics Letters*, vol. 30, pp. 205 – 206, Feb. 1994.

[72] D. Frey and A. Tola, "A state-space formulation for externally linear class AB dynamicalcircuits," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]*, vol. 46, no. 3, pp. 306–314, 1999.

[73] A. Tola and D. Frey, "A Study of Different Class AB Log Domain First Order Filters," *Analog Integrated Circuits and Signal Processing*, vol. 22, no. 2, pp. 163–176, 2000.

[74] C. Enz, M. Punzenberger, and D. Python, "Low-voltage log-domain signal processing in CMOS and BiCMOS," *Circuits and Systems II: Analog and Digital Signal Processing, IEEE Transactions on [see also Circuits and Systems II: Express Briefs, IEEE Transactions on]*, vol. 46, no. 3, pp. 279–289, 1999.

[75] R. Velmurugan, "Implementation strategies for particle filter based target tracking," Ph.D. dissertation, Georgia Institute of Technology, 2007.

[76] R. Velmurugan, S. Subramanian, V. Cevher, D. Abramson, K. M. Odame, J. D. Gray, H. Lo, J. H. McClellan, and D. V. Anderson, "On low-power analog implementation of particle filters for target tracking," in *Proc. 14th European Signal Processing Conf. EUSIPCO 2006*, Sept. 2006.

[77] G. Serrano, P. D. Smith, H. J. Lo, R. Chawla, T. S. Hall, C. Twigg, and P. Hasler, "Automatic rapid programming of large arrays of floating-gate elements," in *Proceedings of the International Symposium on Circuits and Systems*, vol. 1, May 2004, pp. I–373 – I–376.