

BRUSSELS
SCHOOL
OF ENGINEERING

Multi-Objective Optimization and Multi-Criteria Decision Aid
Applied to the Design of 3D-Stacked Integrated Circuits

DOAN Nguyen Anh Vu



Thèse présentée en vue de l'obtention du grade de Docteur en
Sciences de l'Ingénieur et Technologie sous la direction des Professeurs
Yves De Smet, Dragomir Milojevic et Frédéric Robert

Acknowledgements

This thesis would not have been possible without the mentoring of my supervisors Yves De Smet, Dragomir Milojevic and Frédéric Robert. Thank you for your support and advices.

I would also like to thank Pierre Mathys for his involvement in my *comité d'accompagnement* and all the very interesting discussions regarding my thesis.

I am grateful for the opportunity to collaborate with Xavier Gandiseux in Nantes and Bertrand Granado and Andrea Pinna for welcoming me into their lab in Paris.

It was very rewarding and enjoyable to work with and discuss my ideas with my colleagues and friends at the ULB: Michaël, François, Renaud, Karim, Matthieu, Oli, Antoun, Axel, Thierry, the Yannicks. Thank you for the stimulating environment you created.

Where would I be without my friends: Roxane, Lauriane, Morgane, Cédric, Céline, Pierre-Olivier, Niall, Juliette, Benjamin, Nathalie, Guillaume, Momo, Antoine, Loïc and many others that I will inevitably forget. Thank you all for having crossed my path, cheering me up when I needed it and the friendship you honour me with.

My deepest gratitude goes to my family who helped me succeed in my studies and this thesis, with their unconditional love and support.

Finally, a special thank to Lucie for lighting up my life with this universe we share with each other.

Abstract

In the past decades, the microelectronic industry has been following the Moore's law to improve the performance of integrated circuits (IC). However, it will probably be impossible to follow this law in the future due to physical limitations appearing with the miniaturization of the transistors below a certain threshold without innovation. In order to overcome this problem, new technologies have emerged, and among them the 3D-Stacked Integrated Circuits (3D-SIC) have been proposed to keep the Moore's momentum alive. 3D-SICs can bring numerous advantages in the design of future ICs but at the cost of additional design complexity due to their highly combinatorial nature, and the optimization of several conflicting criteria. In this thesis, we present a first study of tools that can help the design of 3D-SICs, using multiobjective optimization (MOO) and multi-criteria decision aid (MCDA). Our study has targeted one of the main issues in the design of 3D-SICs: the partitioning with floorplanning estimation under multiple objectives. This thesis shows that the use of a multi-criteria paradigm can provide relevant and objective analysis of the problem. This can allow a quick design space exploration and an improvement of the current design flows as it is possible to provide qualitative and quantitative information about a design space, that would not be available with current tools. Also, with its flexibility, MOO can cope with the multiple degrees of freedom of 3D-SICs, which enables more design possibilities that are usually not taken into account with current tools. In addition, the developed algorithms can show robustness properties even if the problem is complex. Finally, applying multi-criteria decision aid would allow designers to make relevant choices in a transparent process.

Contents

Acknowledgements	i
Abstract	iii
Introduction	1
Publications	7
1 Review of the literature: <i>Part I: Microelectronics design</i>	9
1.1 Introduction	9
1.2 3D integration	10
1.2.1 Manufacturing technologies	10
1.2.2 3D-SIC advantages	12
1.2.3 3D-SIC design challenges	14
1.3 Current design flows and their limitations	15
1.4 Design space exploration tools	16
1.4.1 2D-IC tools	17
1.4.2 3D-SIC tools	18
1.5 Conclusion	19
2 Review of the literature: <i>Part II: Operations research</i>	21
2.1 Introduction	21
2.2 The uni-criterion paradigm	22
2.2.1 Problem formulation	22
2.2.2 Examples of typical optimization problems	22
2.3 From the uni-criterion paradigm to the multi-criteria paradigm	24
2.4 The multi-criteria paradigm	25
2.4.1 Problem formulation	25
2.4.2 Metaheuristics for multi-objective optimization	26
2.4.3 Multi-criteria decision aid	33
2.4.4 Preference modelling	34

2.4.5	Some important multi-criteria methods	35
2.5	Conclusion	46
3	Problem definition, modelling and simulation results	47
3.1	Introduction	48
3.2	Problem definition	48
3.2.1	Designing an IC	48
3.3	Model and criteria definition	49
3.4	Multi-objective optimization	52
3.4.1	Overview of the method	52
3.4.2	Implementation of the method	54
3.5	Experimental set-up and results	62
3.5.1	Case study 1: basic MPSoc analysis	62
3.5.2	Case study 2: scalability	65
3.6	Robustness of the methodology	73
3.6.1	Contribution indicator	73
3.6.2	Spread indicator	73
3.6.3	Binary ϵ -indicator	74
3.6.4	Unary hypervolume indicator	75
3.6.5	Density of the Pareto front - gaps in the frontier	76
3.6.6	Convexity indicator	78
3.6.7	A word on the scalability of the methodology	78
3.7	Conclusion	78
4	Results exploitation	79
4.1	Introduction	80
4.2	Preference modelling	80
4.2.1	Using the PROMETHEE methods	80
4.3	Constraint modelling	85
4.4	Pertinently representing multi-criteria information in evaluation tables	87
4.5	On the use of a multi-criteria paradigm in microelectronic design . .	87
4.6	Conclusion	88
5	Conclusions	91
Bibliography		95
Appendix		103
I	Scalability case study - evaluation table	104
II	Scalability case study - bandwidth input matrix	121

CONTENTS**vii**

III PROMETHEE-compatible presentations of multicriteria evaluation tables	123
Résumé (French Summary)	159

List of Figures

1	Moore's law [1]	2
2	Trends in transistor gate delay and interconnect delay with IC fabrication technology [2]	2
1.1	Illustration of the wiring properties of a 3D-SIC [3]	11
1.2	Shorter interconnections [4]	12
1.3	Silicon efficiency [4]	13
1.4	Components accessibility [4]	13
1.5	Bandwidth improvement [4]	14
1.6	Classical design flow	16
2.1	Illustration of Pareto ranks	30
2.2	Crowding distance calculation [5]	31
2.3	Example of a GAIA plane	42
3.1	Power grid and thermal map of a partition (3 tiers)	51
3.2	MCDA-based design methodology for 3D DSE partitioning	53
3.3	Example of aspect ratio degree of freedom	53
3.4	The use of different manufacturing technologies results in a size variation	54
3.5	General NSGA-II steps	56
3.6	Evolution of the crossover probability as a function of the distance between two solutions	58
3.7	Architecture of the 3MF MPSoC platform [6]	63
3.8	IL-cost projection view of the Pareto frontier	66
3.9	3D view (interconnection length-volume-cost) of the Pareto frontier	67
3.10	IL-cost projection view of the Pareto frontier (with and without aspect ratio)	68
3.11	Extended 3MF platform (1 layer)	70
3.12	Extended 3MF platform (3 layers)	71
3.13	Evolution of the contribution indicator	74

3.14	Spread indicator I_s function of the neighbourhood parameter σ	75
3.15	Evolution of the unary hypervolume indicator (averaged values compared to the reference set R over time)	77
4.1	GAIA plane of the case study	82
4.2	GAIA plane of the case study (criteria axis only)	83
4.3	GAIA plane of the case study (with decision axis projection)	84
4.4	Constraints modelling (without filtered alternatives)	86
4.5	Constraints modelling (with filtered alternatives)	87

List of Tables

2.1	Data associated with the LP problem	23
2.2	Preference functions (reproduced from [7])	39
3.1	Recap table of the criteria	52
3.2	Output matrix template	55
3.3	First parent, ordered by L column; the line specifies the crossover cut	59
3.4	Second parent, ordered by L column; the line specifies the crossover cut	59
3.5	Possible child, ordered by L column	60
3.6	First parent, ordered by ID column; the line specifies the crossover cut	60
3.7	Second parent, ordered by ID column; the line specifies the crossover cut	61
3.8	Possible child, ordered by ID column	61
3.9	Scenario input matrix example	63
3.10	"Data split scenario" bandwidth requirements	64
3.11	Bandwidth input matrix	64
3.12	Available technologies input matrix example	64
3.13	Scalability case input matrix	67
3.14	Scalability case bandwidth requirements	69
3.15	Evolution of the contribution indicator	73
3.16	Evolution of the binary ϵ -indicator (averaged values compared to the reference set R) over time	75
3.17	Comparison of the binary ϵ -indicators for each experiment	76
3.18	Comparison of the binary ϵ -indicators between iterations of the same experiment	76
3.19	Evolution of the unary hypervolume indicator (averaged values compared to the reference set R) over time	76
3.20	Hypervolume for each experiment	77
4.1	PROMETHEE model	81
4.2	Stability intervals (level 1)	85

Introduction

2D architecture, current design flows and their limitations

In order to continuously improve the performance of integrated circuits (IC), technologists deploy enormous efforts to produce IC manufacturing process that is compelling to follow the well-known Moore's Law (see Figure 1). This empirical law predicts a doubling of the transistors' integration each 18 months and therefore increasing logic capacity of the circuit per unit area.

The improvements of 2D architectures are primarily driven by the reduction of the transistor size. By reducing transistor dimensions, the switching speed is increased thanks to the shorter distance between the source and the drain, implying an improvement of the overall speed of the designs.

However, as the transistor size is decreasing, the observed improvement is also getting smaller. Indeed, a smaller transistor allows higher device density but will slightly increase the total delay (sum of gate and interconnection delays) at the level of the complete circuit. Also, the total power consumption is increased due to higher leakage and increasing interconnection wire length [8]. In Figure 2 is shown the trends in transistor gate delay and interconnect delay with IC fabrication technology where the crossover point represents the interconnect bottleneck. As transistor gate dimensions decrease, designs stretch the limits of the metal interconnection lines that bind the devices, and interconnect performance has become the main limit to IC performance. This constraint is illustrated by the increasing gap between the gate delay (switching speed) of a transistor and the propagation delay between transistors of an integrated circuit [2].

With the miniaturization, quantum effects such as quantum tunnelling will significantly affect how a transistor behave. Indeed, even if a transistor is blocking, current can flow through due to quantum tunnelling such that it will be difficult to control its state and thus the basic working principle of a transistor [9].

In addition to these physical aspects, economical considerations that will hinder the IC evolution beyond 20nm have to be taken into account [8, 10].

In order to overcome these limitations, new technologies have been proposed such as the carbon nanotubes [11], the nanowire transistors [12], the single-electron

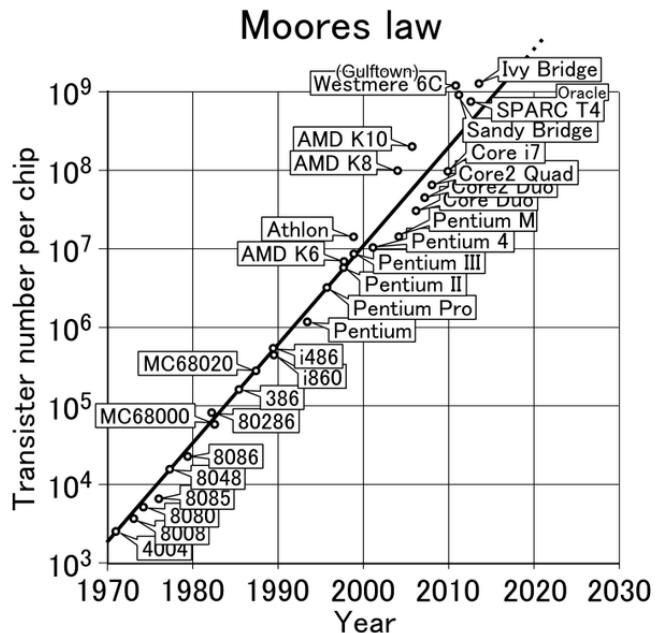


Figure 1 – Moore’s law [1]

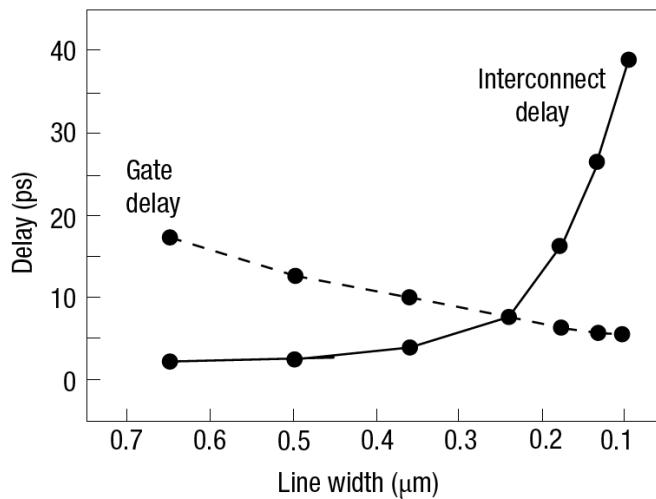


Figure 2 – Trends in transistor gate delay and interconnect delay with IC fabrication technology [2]

transistors [13], but also the 3D-Stacked Integrated Circuits (3D-SIC) proposed by the academic and industrial communities. The latter has been often cited as the most prominent one as it is based on the current technologies and still uses silicon as basis material; 3D-SICs can also allow shorter interconnection lengths, smaller footprint, larger bandwidth, heterogeneous circuits among their main advantages [4, 14–16].

Fast evolution of IC manufacturing technologies makes even the design of 2D-ICs a complex and tedious task with the growing number of design choices at the system level (e.g. number and type of functional units and memories, type and topology of the interconnection system, etc.) and physical level (respecting area/timing/power constraints). Using 3D-SICs introduces even more degrees of freedom: number of tiers, choices for manufacturing technology (e.g. full 3D integration, silicon interposer, face-to-face, back-to-face, etc.), 3D partitioning and placement strategies etc. These new degrees of freedom will contribute to the combinatorial explosion of already huge design spaces. Moreover, practice and 2D design experience cannot be fully exploited with 3D technology, since 3D-SICs change considerably the way ICs are implemented. The current design flows, which already showed their limits with conventional 2D-ICs due to their sequential nature, may thus need improvements to be able to deal with the increased complexity of emerging 3D-SICs [10, 17].

One of the solutions to face this problem is to develop high-level tools which can quickly explore design spaces and give early and reasonably accurate performance estimations based on physical prototyping of the 3D circuits [10].

In addition, performance estimation/optimization and the selection of the most-suitable solutions usually implies to take several objectives into account (e.g. maximization of the performance, minimization of the cost, minimization of the package size, etc.).

Currently, the design tools can be considered to follow a uni-criterion paradigm. Indeed, they have sequential development steps and each criterion is optimized without considering the impact on other criteria. This can lead to several rollbacks in the design flow since the achievement of the requirements can be time consuming (typical design iterations are measured in weeks). For instance, current tools will only minimize the area of a circuit to reach the timing constraints by solving a 2D place and route problem and this will be more complex with 3D-SICs because the system has also to be partitioned.

On the other hand, multi-objective approaches have been developed to optimize all the criteria simultaneously. Designing 3D-SICs inherently implies a huge design space and numerous degrees of freedom and criteria. As it will be shown in Chapter 3, it appears that modelling a 3D circuit is not an easy task since numerous 3D specificities have to be taken into account. Also, defining and optimizing the criteria is not trivial as they are of heterogeneous nature. Modelling this problem is therefore challenging and these are the reasons why we propose to apply a multi-criteria paradigm for the design of 3D-SICs.

Research questions

Multi-objective optimization and multi-criteria decision aid were developed from the need of taking into account several criteria simultaneously. These tools from the operations research field have shown their abilities in solving similar problems in other fields, which also have a large solution space and applying metaheuristics have shown interesting results [18].

In this thesis, we will show the applicability of a multi-criteria paradigm for the design of 3D-SICs:

- how a 3D circuit can be modelled to apply multi-objective optimization: 3D-SICs have numerous specificities and taking them into account in a model is not trivial. This will be developed in Section 3.3.
- how multi-objective optimization can be used to optimize a 3D circuit: the criteria are of heterogeneous nature which can increase the difficulty of optimizing them. This will be presented in Section 3.4.2.
- what kind of information can be provided to a designer: multi-objective optimization can provide qualitative and quantitative information that would not be available with current design tools. This will be illustrated in Section 3.5.1.
- can MOO algorithms deal with the complexity of 3D-SICs: good properties of convergence and diversities can be shown. This will be studied in Section 3.6.
- how multi-criteria decision aid can exploit these results to assist a designer: after the optimization process, making a choice among the possible designs is not trivial since several criteria are considered simultaneously and MCDA can assist a designer facing such situations. This will be introduced in Chapter 4.

Outline

In the first chapter, we will take an overview of the design and manufacturing of 3D-SICs. We will explain the limitations of current design flows and present the developments that have been carried out to overcome these problems. We will discuss why they should be improved and introduce how a multi-criteria paradigm can be useful.

In the chapter two, we will present a short overview of the main tools in the MCDA fields where some of the most-used methods will be presented.

In the third chapter, we will define the problem we tackle (the 3D partitioning with floorplanning estimation) with the considered criteria. We will then show how a 3D-SIC can be modelled in order to apply multi-objective optimization. Simulations will be run on a case study and show what kind of information can be provided to a designer. The methodology will then be validated with a realistic case study to show the added value of a multi-criteria paradigm compared to a uni-criterion approach.

In the chapter four, we will study the robustness of the methodology and the associated algorithms. We will use classical indicators of the fields to analyse the convergence and diversity properties.

In the fifth chapter, we will explain how the obtained results can be exploited using multi-criteria decision aid. We will discuss on how such a paradigm can be used for designing circuits and what needs to be done in order to integrate it to actual design flows.

Finally, we will conclude on the results of the thesis and express some possible perspectives.

Publications

The works described in this thesis have led to several publications in refereed journals and conference proceedings:

Journal papers

- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "A MOO-based methodology for designing 3D-stacked integrated circuits", *Journal of Multi-Criteria Decision Analysis*, vol.21, no. 1-2, pp. 43-63, January-April 2014
- K. Lidouh, N.A.V. Doan, Y. De Smet, "PROMETHEE-compatible presentations of multicriteria evaluation tables", *International Journal of Multicriteria Decision Making*, to be published (minor revision)

Peer-reviewed conference proceedings

- N.A.V. Doan, F. Robert, Y. De Smet, D. Milojevic, "MCDA-based methodology for efficient 3D-design space exploration and decision", *International Symposium on System-on-Chip Proceedings (SOC 2010)*, Tampere (Finland), pp. 76-83, September 2010
- N.A.V. Doan, Y. De Smet, F. Robert, D. Milojevic, "On the use of multicriteria decision aid tools for the efficient design of 3D-stacked integrated circuits: a preliminary study", *Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management (IEEM 2010)*, Macau (China), December 2010
- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "A MOO-based methodology for designing 3D-stacked integrated circuits", *Journal of Multi-Criteria Decision Analysis*, vol.21, no. 1-2, pp. 43-63, January-April 2014
- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "Using the PROMETHEE methodology for the design of 3D-stacked integrated circuits", *1st International MCDA Workshop on PROMETHEE (IMW2014)*, Bruxelles (Belgique), January 2014

- K. Lidouh, N.A.V Doan, Y. De Smet, "PROMETHEE-compatible presentations of multicriteria evaluation tables", *2nd International MCDA Workshop on PROMETHEE (IMW2015)*, Bruxelles (Belgique), January 2015, to be published (accepted)

Presentations at conferences

- N.A.V. Doan, F. Robert, Y. De Smet, D. Milojevic, "Utilisation d'outils multicritères pour la conception efficace de circuits intégrés 3D", *72èmes Journées du groupe de travail européen Aide multicritère à la décision (MCDA72)*, Paris (France), October 2010
- N.A.V. Doan, Y. De Smet, F. Robert, D. Milojevic, "On the use of multi-criteria decision aid tools for the efficient design of 3D-stacked integrated circuits: a preliminary study", *Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management (IEEM 2010)*, Macau (China), December 2010
- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "Multi-Objective Optimization for the Design of 3D-Stacked Integrated Circuits", *Conference of the Belgian Operational Research Society (ORBEL 28)*, Mons (Belgique), January 2014

1

Review of the literature

Part I: Microelectronics design

Chapter abstract

In this chapter, we present a short overview of the 3D-stacked integrated circuits that have been proposed by the industrial and academic communities to overcome 2D-IC's limitations. We show how 3D-SICs can be designed and manufactured and explain why current design flows should be improved to deal with these new challenges.

1.1 Introduction

In the introduction, we have quickly presented the state of the field of microelectronics design where the industry will be facing physical limitations of the silicon. In order to overcome this problem, 3D-stacked integrated circuits have been proposed and by the industrial and academic communities. In this chapter we will present a short overview of this technology: show how 3D-SICs can be manufactured, their advantages and drawbacks, the challenges when designing them. We will then discuss about the limitations of current design flows and why they should be improved in order to address the complexity of designing 3D-SICs.

1.2 3D integration

Most of the current ICs are designed with electronic components (i.e. transistors) that are planar (although multi-gate transistors, such as finFETs tends to extend in the 3rd dimension) interconnected using up to a maximum of 12 (also planar) wiring (metal) layers per circuit. Those conventional ICs can thus be considered to be two-dimensional (2D)-ICs since both device and interconnect are predominantly made in a planar way [19, 20]. As a major evolution of 2D-ICs, 3D-SICs are designed with multiple traditional 2D-ICs (that are manufactured independently, using standard CMOS technology) that are assembled (stacked) vertically in 3D-tiers. In face to back configuration, different 2D circuits communicate between tiers using vertical interconnections that need to connect front side of the chip and the backside, i.e. they need to traverse bulk silicon. In face to face configuration, the chip communicate with their respective front sides. These connections can be Through Silicon Vias (TSV) and if it is in a face to face configuration, micro bumps (μ Bump) or copper pads (Cu-Pad) can also be used. These binding technologies can be today manufactured with satisfactory geometrical properties, namely their diameter, pitch and height, allowing efficient integration of real-world systems [21, 22]. This is shown in Figure 1.1, where 2 dies, oriented face down are connected (Figure 1.1a). An active component (i.e. logic gate) of the T1 is connected to the T2 using a TSV, back side metallization layer (to enable TSV placement anywhere in the T1 die), and μ bump on the top layer of the T2, that is then connected, through a series of metal layers of the T2, to the active component of the top tier (T2). The corresponding face to face configuration is shown in Figure 1.1b.

1.2.1 Manufacturing technologies

Several 3D manufacturing technologies have been proposed and have been used to implement complete systems. Among the existing possibilities, four major categories of methods that illustrate 3D integration can be cited [4, 14].

Transistor stacking The transistor stacking consists in creating several transistors level on one substrate. This should be the better way to manufacture 3D circuits although the success rate are currently limited due to thermal issues among the different limitations. The required temperatures to create a layer of high-performance transistors would provoke the destruction of the copper and aluminium already laid down on the previous layer [4].

Chip stacking This methods consists in stacking components that have been designed and tested separately to produce a system-in-package (SiP). The vertically-stacked chips are interconnected with traditional wirings (lateral wire bondings). The

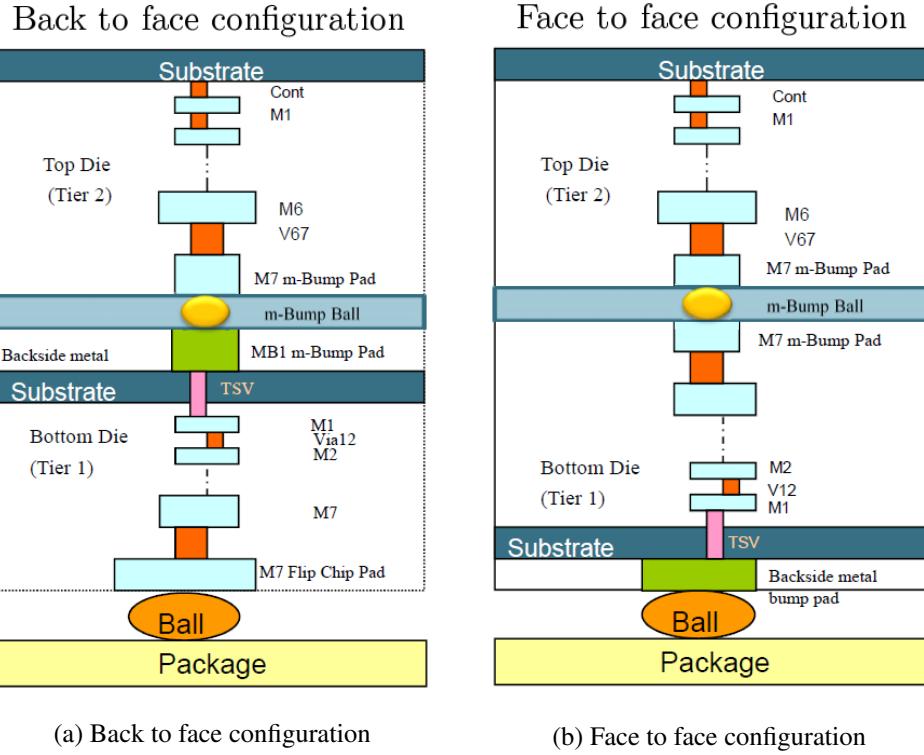


Figure 1.1 – Illustration of the wiring properties of a 3D-SIC [3]

principal advantage of this method is an improvement in terms of size. The wirings are shorter however the components integration density is not increased compared to a 2D system [4].

Die-on-wafer stacking In this method, known good dies (KGD), which are functional tested chips, are connected to a host wafer containing other KGDs. These KGDs can be interconnected with organic glues, oxide or metal bonding. The wafer and the bonded KGDs are then shaped to create the interconnections. Different substrates can be combined if the required temperature is low enough to minimize non-homogeneous expansion effects [4].

The die-on-wafer stacking can use interconnections on the edges of the chips or through-die. Depending on the interconnection type, this method can produce a better integration level than the chip stacking, with a better cost per connection ratio and a higher interconnection density, while holding the advantages of the KGDs.

The quality of the stacking depends on the pick-and-place equipment which is used to position the dies on the wafer. The placement accuracy will determine the

possible interconnection density. Also, current equipments are supposed to handle fully buffered chips, not naked circuits so it does not provide protection to static discharge [4].

Wafer-level stacking This methods consists in bonding entire wafers into a stack. The vertical through-wafer connections are made directly trough each substrate to the next wafer and its transistors layer. Similarly to the previous method, the interconnection density rely on the precision of the alignment, which is however currently better than the die-on-wafer stacking. This greater accuracy implies a better cost per connection ratio and a higher interconnection density compared to the die-on-wafer stacking [4].

The use of mixed substrates is also possible, only limited by the process temperatures. All the processing is done at the wafer level so wafer handling equipments are used. Since these provide protection to static discharge so there is no need to include buffering between the layers. The methods to bind two wafers are the same that are available for the die-on-wafer method.

One drawback to wafer-level stacking is its efficiency, since the chips on a wafer are not all KGDs.

1.2.2 3D-SIC advantages

Interconnection length The 3D integration allows to design circuits with components closer to each other. Wire of a few millimetres long can be replaced by TSV of a few tens of microns, as shown in Figure 1.2. These shorter interconnections will introduce shorter delays, hence allowing higher working frequencies [4, 15].

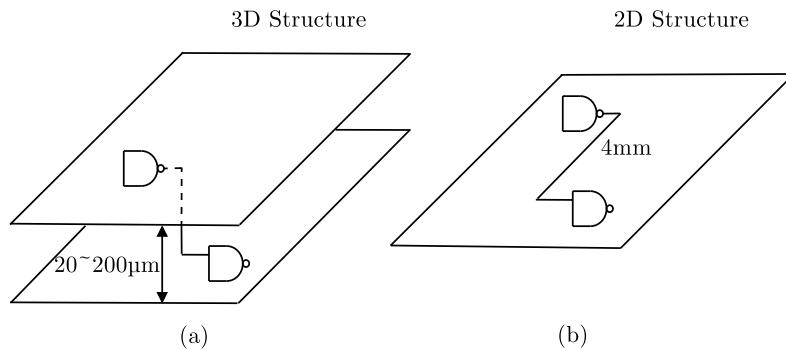


Figure 1.2 – Shorter interconnections [4]

Silicon efficiency and accessibility Adding a vertical dimension allows to increase the integration density. It is therefore possible to have more logic gates than a 2D-IC

for the same footprint, hence a more efficient use of the silicon as shown in Figure 1.3. For instance, compared to the footprint of a 2D-IC, the 3D-SICs can double the integration for a 50% use of a 2D footprint [4].

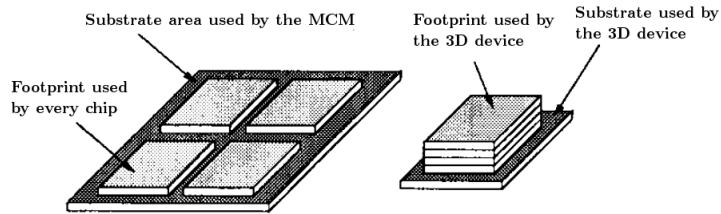


Figure 1.3 – Silicon efficiency [4]

In addition, the 3D integration allows a better accessibility for the components, as shown in Figure 1.4. Indeed, for a 2D structure, 8 accessible neighbours can be considered for a central element (Figure 1.4a), whereas for a 3D structure, the number of accessible neighbours can reach 116 with through-tiers interconnections (Figure 1.4b) [4].

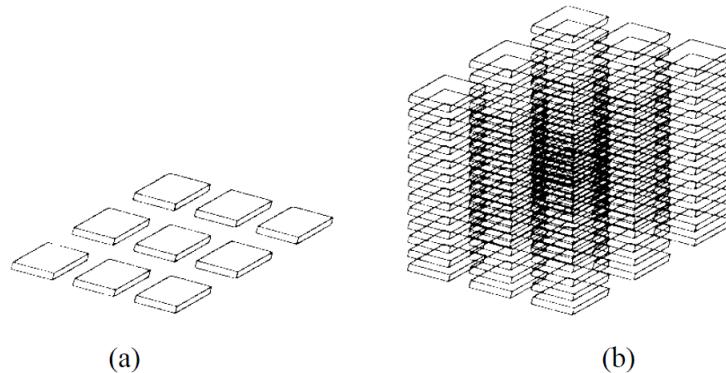


Figure 1.4 – Components accessibility [4]

Bandwidth The use of TSVs on 3D-SIC can significantly increase the bandwidth of a circuit. Indeed, as shown in Figure 1.5, the interconnections are not only limited to peripheral connections but can also make use of the circuit's surface. At a same working frequencies, this allows more bandwidth while at lower frequencies, the same bandwidth usage will require less power.

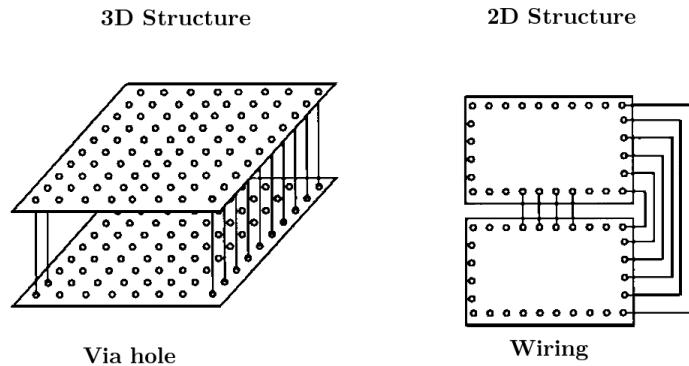


Figure 1.5 – Bandwidth improvement [4]

Consumption and noise Shorter interconnections generally translates into lower capacitance and inductance parasitics. This means a decrease of the numbers of repeaters, hence a better consumption, less noise and less jitter hence lower delays and power consumption.

Heterogeneous circuits The 3D technologies allow truly heterogeneous designs. For instance, it is possible to integrate, in addition to traditional digital circuits of different technologies, analogical circuits such as sensors or antennas, as well as power supply, which give 3D-SIC a high degree diversity. It is also possible to mix old and new generations technologies to lower the cost of future circuits. [16].

1.2.3 3D-SIC design challenges

As explained, 3D-SICs offer numerous design perspectives thanks to their advantages. However there are drawbacks that need to be taken into account and that will be discussed in the following paragraphs.

Thermal dissipation The power density has increased exponentially over the past decades for the 2D-ICs and it appears that this trend will continue in the near future. As for 3D-SICs, due to their higher component density, they will also be subject to higher power density so thermal management should be considered carefully [4]. A simplified model of thermal dissipation has been developed in this thesis and will be presented in Chapter 3.

Cost With the appearance of a new technology, the involvement of a high cost should often be expected. In the case of 3D technology, in addition of the cost of the

technology itself, there are the lack of infrastructure and the reluctance of manufacturers who do not want to risk to change to new technologies [4].

Design complexity and design software A large number of systems have been implemented using the 2D technologies which means that current tools can cope with 2D design complexity even if they show more and more their limits [10, 17]. As for 3D-SICs, the increased complexity can be tackled by developing adapted software [4]. However, to the best of our knowledge, few 3D dedicated software currently exist. One can nevertheless cite the works in [23–26]. Most of other tools are mainly developed for and owned by particular manufacturers. In addition, they are based on 2D design tools which does not allow to tackle the complexity of 3D designs integrally as they do not take all 3D specificities into account. For example, as it will be shown in Section 1.3, 3D partitioning and floorplanning are considered separately (as it is done for 2D-ICs) whereas the 3D geometrical assignment should be taken as a whole.

In the following section, we will have an overview about these software tools and generally about the design flow used to design integrated circuits.

1.3 Current design flows and their limitations

Design flows are the combination of electronic design automation (EDA) tools used to produce an integrated circuit. These flows can generally be summarized in 4 main steps [27], as shown in Figure 1.6.

As one can observe, the design flows are sequential. The process goes from one step to the other with local optimization loops. In practice, it is not unusual to have several rollbacks to the previous steps due to the need of manual intervention to adjust the constraints and reach good solutions.

As explained previously, designing ICs implies numerous choices. At the moment, with this growing complexity, the current design flows can already show their limits. Indeed, these tools are developed to only take into account particular specificities of a circuit (area, timing) whereas other characteristics also have to be optimized. Besides, most of the time, the designers will be likely to freeze a certain amount of choices on basis of their experience, and then begin the optimization process with the remaining parameters. This will therefore limit the exploration of the design space and other good solutions may be ignored. In addition, the fixed choices can be questionable since they are based on the designer's experience and it is difficult to master the whole complexity of the problem in terms of criteria and possibilities of design.

The current design flows, which already showed their limits with conventional 2D-ICs, may thus need improvements to be able to deal with the increased complexity of emerging 3D-SICs [10, 17].

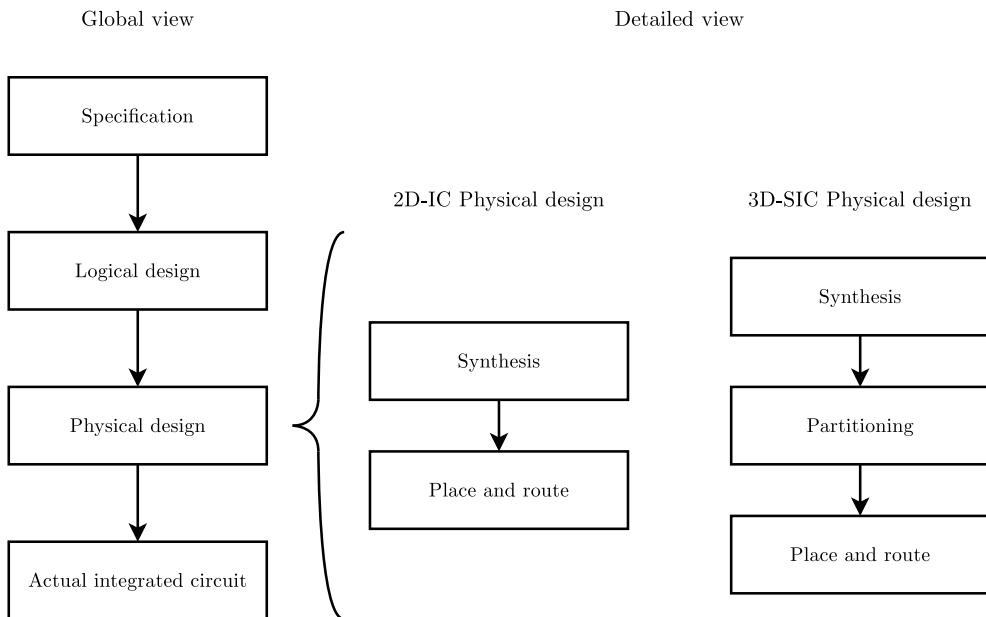


Figure 1.6 – Classical design flow

For the moment, most 3D design flows adapt classical flows to include 3D specificities, in particular 3D partitioning and 3D place & route (see Figure 1.6). We can observe that these two steps are separated: the circuits are first (manually) partitioned, then the place & route occurs for each layer. However, one can guess that the performances of a 3D-SIC will depend on the position of a component, considering simultaneously its position on a layer ((X,Y)-coordinates) and the layer where it lies (Z-coordinate). 3D design flows therefore need improvements to take into account these three coordinates at the same time.

1.4 Design space exploration tools

In order to cope with the increasing complexity of integrated circuits and the limitations of the current design flows, numerous tools have been proposed, in particular works about design space exploration (DSE) that have been developed to quickly suggest possible interesting solutions to a designer and speed up the design processes. In this section, we will describe different DSE tools that have been proposed in the literature.

1.4.1 2D-IC tools

MILAN The MILAN (Model based Integrated simuLAtioN) framework [28] aims to simplify the optimization and the exploration of design spaces for System-on-Chip (SoC) platforms. This tool works on the component level and allows the users to choose a compromise between the simulation speed and the results accuracy. The exploration and optimization process is done in two phases: first it searches for possible combinations between the architecture, the application and the mapping and second it estimates the performances (power, latency) depending the precision asked by the users.

SoC Architecture Explorer SoC Architecture Explorer [29] is a multi-objective optimization and exploration tool that aims the design of SoC architectures by evaluating the compromises between the footprint and the execution time. The exploration process focuses on the application and the system architecture where the tool analyses the data flow and estimates the data transfers to determine a number of possible architectures.

modeFRONTIER (ESTECO) modeFRONTIER [30] is a proprietary development environment developed by ESTECO. It is a multi-objective optimization tool that aims parallel SoC architectures. modeFRONTIER allows to deal with up to one million different design configurations thanks to statistical analysis tools and data mining techniques.

MULTICUBE The MULTICUBE project (MULTI³) [31,32] is a European project started in 2008 and dedicated to the multi-objective exploration of MPSoC architectures for multimedia embedded systems. The aim is to develop a framework that allows a quick and automated exploration of the design space to improve the performances of a MPSoC with metrics such as power, latency, computing performance, bandwidth, QoS, etc. This project is based on several heuristics and optimization algorithms that reduce the exploration time and allow a quick selection of the best solutions. In addition, MULTICUBE also aims to define an application-oriented framework based on the results of the multi-objective exploration to optimize the resources allocation and the tasks scheduling of the applications. The exploration is done at the system level, using the SystemC language. The project includes proprietary and open-source tools whose development targets the industry. Among the developed prototyping tools, Multicube explorer and Multicube-SCoPE can be cited.

Multicube Explorer Multicube Explorer [33] is a design space exploration framework for supporting platform-based design. This tool allows a fast optimization of a system with objective functions such as power, delays, surface, etc. by means of

a system simulator. Multicube explorer proposes several multi-objective optimization methods that aim to propose the best compromises.

Multicube-SCoPE Multicube-SCoPE [34] is an evolution of the SCoPE tool [35] oriented to design space exploration. It is a fast system performance and power simulator providing metrics associated with a system in order to drive the DSE process.

1.4.2 3D-SIC tools

DSE for 3D-stacked DRAMs by Weis *et al.* Design space exploration for 3D-stacked DRAMs has been developed by Weis *et al.* [36]. They defined a 3D-DRAM based on a SystemC model with a 3D channel controller and also considered a wiring model for the TSVs. The used metrics are area, performance and energy efficiency evaluated for different DRAM architectures and technologies. 3D thermal issues has been kept out of the scope of the study. The simulation results allowed them to have a trade-off analysis of horizontal wirings against vertical wirings in terms of energy and cell efficiency. They could show quantitatively how a 3D-DRAM can perform better than a classical DRAM.

Observation This work is really interesting as it shows the stakes of using the 3D technology for DRAM. However, since it is based on DRAMs, the tools work with a memory structure that is repeated in the 3D-DRAM, which does not take into consideration more heterogeneous architectures. Also, only trade-off analyses are performed, which does not give a more global multi-criteria insight of the results as it will be illustrated in Chapter 2.

DSE for 3D architecture and DSE for 3D integrated circuits by Xie *et al.* Design space exploration for 3D architecture and design space exploration for 3D integrated circuits are two works proposed by Xie *et al.* [25, 26]. In the first study, they combine several tools to perform a DSE:

- for the 3D cache partitioning, two strategies have been proposed at the subarrays granularity level
- the area, the delay and the energy of a 3D cache are assessed following a cost function
- 3DCacti, a tool developed to explore various 3D partitioning options of caches
- thermal-aware 3D floorplanning based on simulated annealing

With the DSE, they are able to propose different possible architectures for 3D microprocessor design by performing trade-off analyses of the criteria. The second study is an extension where a cost analysis is added.

Observation These works seem to be among the most integrated study in the literature with cache partitioning and microprocessor floorplanning, and considering several criteria including thermal issues. However, the partitioning and the floorplanning are separated while a more 3D approach should consider both dimensions simultaneously. Also, the criteria are aggregated with a cost function which can lead to inconsistency as it will be explained in Chapter 2 and only trade-off analyses are performed.

Automated design flow for 3D microarchitecture evaluation by Cong *et al.* An automated design flow for 3D microarchitecture evaluation has been proposed by Cong *et al.* [24]. They propose an evaluation flow for performance assessment and thermal management. This allows them to perform thermal-aware 3D floorplanning.

Observation This work is worth mentioning as it proposes a quick way to evaluate temperature issues. However, it only deals with the thermal criterion.

PathFinding flow The PathFinding flow is a project proposed by IMEC and Atrenta [37, 38] to deal with design exploration of advanced packaging integration. The aim of this work is to be able to produce a specification for the architecture and for the technology with assessment of performance, power and cost. The methodology is divided in 3 steps:

1. 3D system level design exploration with a rough estimation of the performance, power and cost parameters. The designer will be able to focus on the 2D design issues while manually considering the 3D specificities.
2. RTL (Register Transfer Level) elaboration, which links the system level to the physical design by producing RTL models.
3. 3D physical design prototyping, which allows fast exploration of the physical design impact of alternative design/technology options on the performance, power and cost parameters.

Observation This work is also among the most integrated study in the literature. However, the criteria optimization is done following a uni-criterion approach which does not allow to explore quickly several possibilities.

1.5 Conclusion

In this chapter, we have presented a short overview of the evolution of IC design. Manufacturers have pushed back the limitations of the silicon for the past decades and are now facing new challenges due mainly to quantum effects. 3D-SICs have been

proposed to face these problems and we have shown a quick review of this promising technology.

With the 3D integration, design flows have evolved and integrate 3D partitioning and 3D place and route. However, these two steps are performed separately while they should be considered simultaneously as the circuits' performances will depend on the position of a component on a layer and the layer where it lies.

We have then presented researches that aim to deal with these challenges by making use of multi-objective optimization. To the best of our knowledge, all these tools use a uni-criterion approach or deal with a limited set of criteria while performing only trade-off analyses. This therefore lacks a more global view on all the criteria simultaneously and interesting information about a design space could be missed. The goal of this research is to show that a more multi-objective-oriented optimization could be more suitable to take into account the many aspects of a design.

To achieve this, we will present in the next chapter the tools related to the multi-criteria field. We will then detail how a 3D-SIC can be modelled to apply multi-objective optimization and show that a more multi-criteria is able to provide qualitative and quantitative information that would not be available with current tools.

2

Review of the literature

Part II: Operations research

Chapter abstract

In this chapter, we briefly present the basics of multi-objective optimization and multi-criteria decision aid, in order to justify our choice to use such a paradigm. We will explain in what kind of context a multi-criteria approach can be used and present some classical methods of the field.

2.1 Introduction

In this chapter, we will briefly present the basics of multi-objective optimization and multi-criteria decision aid, in order to justify our choice to use such a paradigm. As stated in Chapter 1, the 3D integration can offer new perspectives but designing 3D-SICs includes two major distinctive features: multiple criteria and a huge number of possible solutions. When facing such problems, two main approaches exist: the uni-criterion paradigm and the multi-criteria paradigm. For optimization problems, these paradigm will refer to the terminology mono-*objective*/multi-*objective* optimization while for decision aid, the terminology uni-*criterion*/multi-*criteria* will be used.

In the following, we will briefly describe each paradigm, showing some of the main approaches alongside illustrative examples. We will first present the uni-criterion methodology, show why it can be limited in our context and explain why a multi-criteria paradigm can be more suitable.

2.2 The uni-criterion paradigm

2.2.1 Problem formulation

An optimization problem can be formulated, without loss of generality, as [7]

$$\begin{aligned} \min f(x) \\ x \in A \end{aligned} \tag{2.1}$$

where f is a real-valued function evaluating the solutions denoted x , and A is the set of solutions, f is also called the *criterion* on which x is evaluated. A criterion can therefore be defined as follows:

Definition 2.1 (Criterion). *A criterion is a way to evaluate and compare potential solutions. These are evaluated according to a specific point of view, which then results in performance levels.*

Let us note that the equation 2.1 expresses a *minimization* problem. A *maximization* problem can be seen as a minimization problem with the identity

$$\max_{x \in A} f(x) = -\min_{x \in A} (-f(x)) \tag{2.2}$$

so that there is no loss of generality by using only *minimization* formulation.

In order to give a more precise idea of what an optimization problem is, we will describe in the next section some typical examples taken from the reference book [7, 18].

2.2.2 Examples of typical optimization problems

Linear programming

Linear programming (LP) is a problem formulation where the aim is to optimize a linear function, subject to linear inequality constraints. This can be formulated as follows:

$$\min \mathbf{c}^T \mathbf{x} \tag{2.3}$$

subject to

$$\begin{aligned} A\mathbf{x} &\leq \mathbf{b} \\ \mathbf{x} &\geq \mathbf{0} \end{aligned}$$

where \mathbf{x} is a vector of continuous, integer or boolean variables to be determined, \mathbf{c} and \mathbf{b} are vectors of coefficients, A is a matrix of coefficients.

Efficient exact methods for solving LP problem exist such as, among the most knowns, the simplex algorithm [39] or the interior point method [40].

Example 2.2 (Linear programming). A given company produces two electronic boards $Board_1$ and $Board_2$ based on two kinds of memories M_1 and M_2 . The objective consists in finding the most profitable product mix, given the availability of each memory M_1 and M_2 , and the amount of memory used as well as the profit per board, as shown in Table 2.1. The decision variables are x_1 and x_2 that represent respectively the amount of $Board_1$ and $Board_2$ produced. The objective is to maximize the profit. The problem can be formulated as an LP:

$$\max \text{ profit} = 5x_1 + 4x_2 \quad (2.4)$$

■

subject to the constraints

$$\begin{aligned} 192x_1 + 128x_2 &\leq 1024 \\ 32x_1 + 64x_2 &\leq 192 \\ x_1, x_2 &\geq 0 \end{aligned}$$

Table 2.1 – Data associated with the LP problem

	Usage for $Board_1$	Usage for $Board_2$	Availability
M_1	192	128	1024
M_2	32	64	192
Profit per unit	€5	€4	

When the variables are restricted to integers, the problems are solved with integer linear programming (ILP):

$$\min \mathbf{c}^T \mathbf{x} \quad (2.5)$$

subject to

$$\begin{aligned} A\mathbf{x} &\leq \mathbf{b} \\ \mathbf{x} &\in \mathbb{N} \end{aligned}$$

where \mathbf{c} and \mathbf{b} are vectors and A is a matrix of coefficients.

When the set of decision variables contains both discrete and continuous variables, the problem refers to **mixed integer programs** (MILP).

Other particular ILP problems which deals with variables that are restricted to be either 0 or 1 are called **0-1 linear programming**.

Example 2.3 (Travelling salesman problem (TSP) [18]). This is one of the most known optimization problem. It can be formulated as follows: given n cities and the distance between each pair of cities, we have to find the shortest tour that visits each city once and returns to the origin city. This problem can be formulated as an

ILP problem.

Let d_{ij} be the distance between the city i and the city j , S be the set of solutions (tours) and define:

$$x_{ij} = \begin{cases} 1 & \text{if the path goes from city } i \text{ to city } j \\ 0 & \text{otherwise} \end{cases}$$

The ILP formulation is then:

$$\min \sum_{i=0}^n \sum_{j=0}^n d_{ij} x_{ij} \quad (2.6)$$

s.t.

$$\begin{aligned} \sum_{i=0, i \neq j}^n x_{ij} &= 1 & j &= 0, \dots, n \\ \sum_{j=0, j \neq i}^n x_{ij} &= 1 & i &= 0, \dots, n \\ \sum_{i \in S, j \notin S} x_{ij} &\geq 1 & \forall S &\subset \{1, \dots, n\} \\ 0 \leq x_{ij} \leq 1 & & \forall i, j \\ x_{ij} \in \mathbb{N} & & \forall i, j \end{aligned}$$

■

Non-linear programming

Non-linear programming (NLP) models deal with mathematical problems where some of the constraints and/or the objective function are non linear:

$$\min f(x) \quad (2.7)$$

where

$$\begin{aligned} f : \mathbb{R}^n &\rightarrow \mathbb{R} \\ x &\in \mathbb{R}^n \end{aligned}$$

subject to

$$g_i(x) \leq 0, i \in J = 1, \dots, m$$

where $g_i : X \rightarrow \mathbb{R}^n$ are the inequality constraints.

NLP are generally more difficult to solve than LP [18] and metaheuristics (see Section 2.4.2) are commonly used to solve this class of problems.

2.3 From the uni-criterion paradigm to the multi-criteria paradigm

With a uni-criterion paradigm, the optimization of one criterion is generally performed while considering that this single criterion synthesizes all the characteristics

of the problems or that the other criteria already satisfy an acceptable level by considering them as constraints. This methodology will search for a solution which is supposed to be optimal according to this criterion. However, most problems encountered in the field of IC design, and more generally in other industrial fields, contains several conflicting criteria as it will be illustrated in Chapter 3. Finding a solution that simultaneously optimizes all the criteria is only possible in rare cases and if optimality can be reached.

For instance, when designing ICs, a manufacturer will try to simultaneously maximize the performance while minimize the cost of the circuit. However, we can already guess that those two objectives are conflicting. Also, producing high-end ICs can be subject to more difficulties in terms of thermal dissipation. In addition, a criterion based on ecological standards may have impacts on the cost and the performance of an IC.

This example shows that a uni-criterion approach cannot always be applied since there is no achievable optimum, as several criteria have to be simultaneously taken into account. A solution that optimizes one criterion will likely affect another.

In order to deal with the multiple criteria of a problem, another paradigm consists in taking into account all the criteria simultaneously: the multi-criteria paradigm.

2.4 The multi-criteria paradigm

2.4.1 Problem formulation

A multi-criteria problem can be formulated without loss of generality as follows [7]:

$$\min_{x \in \mathcal{A}} \{f_1(x), f_2(x), \dots, f_m(x)\} \quad (2.8)$$

where $\{f_1(x), f_2(x), \dots, f_m(x)\}$ is a set denoted \mathcal{F} of m evaluation criteria that needs to be minimized and x is a solution of the set $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$.

As explained in Section 2.3, an optimal solution can be impossible to find for a multi-criteria problem. However, compromise solutions can exist and in order to identify them, a dominance relation can be used which has been defined as follows [7]:

Definition 2.4 (Dominance). A solution a_1 dominates a solution a_2 if:

- a_1 is at least as good as a_2 on all criteria;
- a_1 is strictly better than a_2 on at least one criterion.

From this dominance relation, it is then possible to filter the solutions in order to keep only the non-dominated ones. This set of *efficient* solutions is called the Pareto

frontier. Let us note that the *efficient* solutions refer to the decision space while the Pareto frontier refers to the evaluation space.

Two approaches can be used to establish this set [41]:

- *Exact methods* which aims to compute the Pareto frontier directly [42, 43].
- *Approximate methods* which aims to approach as best as possible the Pareto optimal frontier; a common way is to quickly explore the solution space, for example with metaheuristics [18].

As explained in Chapter 1, designing 3D-SICs involves a huge solution space to deal with in the optimization process. The solution (that is to say the most-suitable 3D-SIC architecture) is unknown and an exhaustive search would take a prohibitive time. Also, due to the nature of the criteria (discrete and continuous variables, linear and non-linear criteria) that will be defined in Chapter 3, we have few hopes to be able to develop an exact method. For those reasons, approximate methods with metaheuristics for multi-objective optimization will be used. Let us also remind that the aim of this thesis is to evaluate the applicability of a multi-criteria paradigm to the design of 3D circuits. Therefore developing exact methods has been kept out of the scope of this work.

2.4.2 Metaheuristics for multi-objective optimization

Metaheuristics are a family of approximate optimization methods. They aim to provide "acceptable" solutions in reasonable time for solving complex problems [18]. As stated previously, finding the optimal solution for a multi-objective optimization problem (MOP) is only possible in exceptional cases since encountering conflicting criteria is generally common. Therefore, multi-objective metaheuristics will not produce a single solution but a set of alternatives defined as Pareto optimal solutions. The main goal is then to obtain this set.

In our study, due to the heterogeneous nature of the criteria, there are few hopes to find the exact Pareto optimal solutions without performing an explicit enumeration. In such cases, metaheuristics are commonly used and the goal is then to find an approximation of this set. Two properties have to be respected in order to ensure good approximations: convergence to the Pareto optimal front and uniform diversity. The first property allows to have solutions that are closed to the Pareto set whereas the second property shows a good distribution around the Pareto front.

Numerous metaheuristics have been developed since the 50s. Among the most known, let us cite genetic algorithm [44], scatter search [45], simulated annealing [46], tabu search [47], memetic algorithms [48] and ant colony optimization [49].

In this work, we will focus on genetic algorithms (GA) as they are quick to implement for a first approach and are suitable to heterogeneous variables problems. More details about other metaheuristics can be found in reference books such as [18, 50, 51].

General description of genetic algorithms

Genetic algorithms have been developed by Holland in the 1970s [44]. They are metaheuristics that reproduce the properties of a natural selection process as described by Charles Darwin. GAs are based on the principle of improvement of gene pool of a population over generations. GAs will mimic the natural evolution with techniques such as selection, crossover and mutation. In the following, we will briefly describe the general methodology of a GA without considering a multi-objective case since the key steps are similar. Afterwards, we will describe one of the most popular multi-objective genetic algorithms: NSGA-II (Non-dominated Sorting Genetic Algorithm) [5].

Genetic algorithms rely on a population that is evolved toward better solutions or individuals. The evolution is an iterative process and starts usually with randomly-generated solutions. At each iteration, every individual is evaluated to define its fitness. The fitter ones are more likely to be selected for genetic modifications (crossover and possibly mutation). The produced solutions constitute the new generation that will be used for the next iteration. The algorithm is commonly terminated when a maximum number of generations has been produced or when no better solution has been produced after a several iterations. The general pseudo-code for genetic algorithms is shown in Algorithm 1.

Algorithm 1: General pseudo-code for genetic algorithms

```

1 CHOOSE initial population;
2 EVALUATE each individual's fitness;
3 repeat
4   | SELECT parents;
5   | CROSSOVER pairs of parents;
6   | MUTATE the resulting offspring;
7   | EVALUATE the new candidates;
8   | SELECT individuals for the next generation;
9 until TERMINATION CONDITION satisfied;
```

Representation of a solution The representation or encoding of a solution is called a chromosome and depends on the problem. Several examples of problems show binary encodings however, in our study we will use a real-valued matrix that will be detailed in Chapter 3. Nevertheless, without loss of generality, we will illustrate the principles of a genetic algorithm by using binary-coded solutions.

Initialization Initially many solutions are generated, usually randomly to form the initial population. Depending on the problem, the generation of the initial population can be guided (seeded) to areas where optimal solutions are likely to be found. Another way to generate solutions can be to use building mechanisms, for instance with a greedy-like algorithm.

Selection The selection is a stochastic process usually planned so that the fitter solutions have a higher probability of being selected. This aims to ensure the convergence of the algorithm.

In particular, one can mention the roulette wheel selection method where the fitness level is used to associate a probability of selection to each candidate. If f_i is the fitness of the individual i , its probability to be selected is $p_i = \frac{f_i}{\sum_{j=0}^n f_j}$ where n is the number of individuals in the population.

Crossover Once a pair of individuals has been selected, they will be crossed-over. Typically, two children are created from each set of parents. One method of crossover (one-point crossover) will be explained here but other approaches exists such as the two-point crossover or the uniform crossover [52]. The one-point crossover is chosen here as it is one of the best performing operators for various kinds of problem and finding a best overall operator is often difficult [53]. A random crossover point will be selected on both parents. Beyond that point, the data will be swapped with the information of the other parent as show in Example 2.5.

Example 2.5 (Crossover example). Let us consider two individuals x and y of the population:

$$\begin{array}{rcccccccccc} x & = & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ y & = & 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0 \end{array}$$

If the randomly-chosen crossover point is 2 then the obtained offspring is:

$$\begin{array}{rcccc|cccc} x' & = & 0 & 1 & | & 0 & 0 & 1 & 0 & 1 & 0 \\ y' & = & 1 & 1 & | & 1 & 0 & 1 & 1 & 0 & 0 \end{array} \quad \blacksquare$$

Mutation Mutation is a genetic operation used to ensure diversity in the generated populations. It changes one piece or more information in the chromosome of an individual. This alteration depends on how the solution is encoded. If it is a bit

string, the most common operation is to apply a bit flip (see Example 2.6) while for float chromosomes, new values can be generated following user-defined rules (see detailed illustration in Chapter 3).

Example 2.6 (Mutation example). Let us consider one individual x' of the population:

$$x' = 0 \ 1 \ 0 \ 0 \ 1 \ 0 \ 1 \ 0$$

■

If the randomly-chosen mutation point is 3 then x' becomes:

$$x' = 0 \ 1 \boxed{1} \ 0 \ 1 \ 0 \ 1 \ 0$$

Termination The generational process is repeated until a termination condition has been encountered. Common conditions are:

- no better results produced after several generations
- fixed number of generations reached
- simulation elapsed time reached

As we can see, when implementing a genetic algorithm, several choices have to be made in terms of methods (crossover type, mutation type, termination conditions) and in addition, the related parameters have to be defined and eventually tuned.

Multi-objective genetic algorithm: NSGA-II

While the original genetic algorithms have been developed for mono-objective purposes, they have also been extended to multi-objective optimization and among the most known, one can cite NSGA-II.

NSGA-II stands for Non-dominated Sorting Genetic Algorithm and has been developed by Deb [5] to provide a multi-objective version for genetic algorithms. It is an evolution of the original NSGA proposed in [54]. NSGA-II follows the same steps as a classical GA and additionally implements techniques, particularly in the selection step, to take into account several objectives simultaneously.

NSGA-II selection The selection is based on the Pareto dominance principle, particularly the Pareto rank which allows to sort all the solutions of a set following an extended Pareto principle and the crowding distance which estimates how dense the surrounding of a solution is.

Definition 2.7 (Pareto rank [5]). From a given pool of solutions, the Pareto optimal ones are of rank 1. For the higher ranks the following process is repeated iteratively: to find the solutions of rank $i \geq 2$, the solutions of rank $i - 1$ are removed and the Pareto solutions from this subset are of rank i . An illustration of the Pareto ranks is given in Figure 2.1.

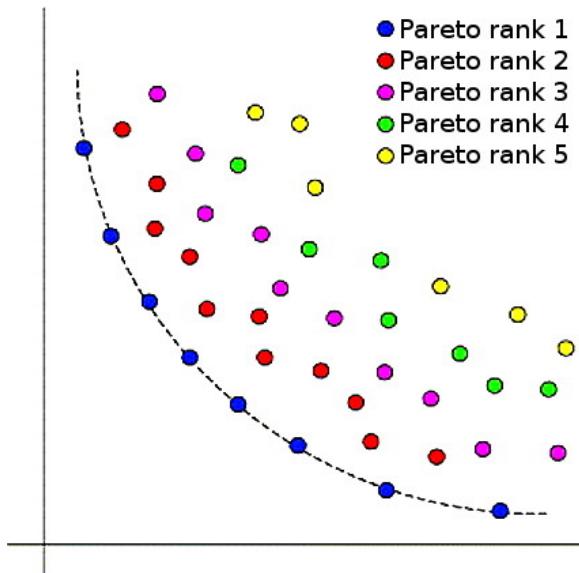


Figure 2.1 – Illustration of Pareto ranks

Definition 2.8 (Crowding distance [5]). The crowding distance is a measure of how close a point is to its neighbours, and thus of the density of solutions surrounding a particular point in the population. It is computed by taking the average distance of the two points on either side of this point along each of the objectives (see. Figure 2.2 and Algorithm 2).

The number of solutions per generation is fixed as constant. Between two solutions with different Pareto ranks, the lower rank will be preferred. Otherwise, if both solutions have the same Pareto rank then the one located in a less-crowded region will be preferred.

Performance evaluation of a metaheuristic

In order to evaluate the performance of a metaheuristic, several metrics have been defined in the literature to assess the convergence and diversity properties of an algorithm [18]. Among the classical indicators, one can cite the contribution indicator,

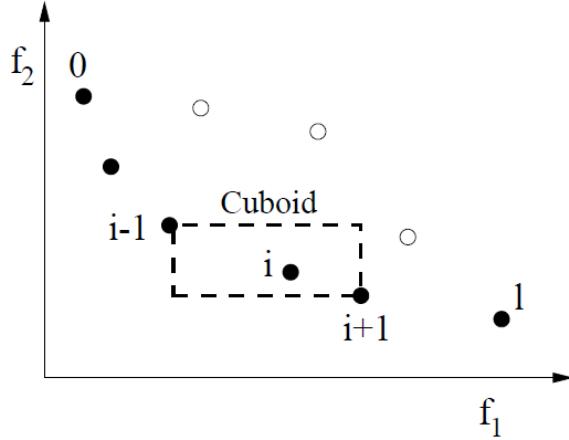


Figure 2.2 – Crowding distance calculation [5]

Algorithm 2: Crowding distance for the set of solutions A [5]

Input: A

- 1 $l = |A|;$
- 2 **foreach** i **do**
- 3 | set $A[i].distance = 0;$
- 4 **end**
- 5 **foreach** objective m **do**
- 6 | $A = \text{sort}(A, m);$
- 7 | $A[1].distance = A[l].distance = \infty;$
- 8 | **for** $i = 2$ to $(l - 1)$ **do**
- 9 | | $A[i].distance = A[i].distance + (A[i + 1].m - A[i - 1].m)$
- 10 | **end**
- 11 **end**
- 12 **Note:** $A[i].m$ refers to the m -th objective function value of the i -th individual in the set A

the spread indicator, the binary ϵ -indicator, the unary hypervolume indicator and the density of the Pareto-front which are presented in [18, 55]. These indicators can be grouped in 3 categories defined in [18]:

1. The convergence-based indicators:

"The convergence metrics evaluate the effectiveness of the solutions in terms of the closeness to the optimal Pareto front."

2. The diversity-based indicators:

"Diversity indicators measure the uniformity of distribution of the obtained solutions in terms of dispersion and extension. In general, the diversity is researched in the objective space."

3. The hybrid indicators: that combine both convergence and diversity measures.

Contribution indicator The contribution is a convergence-based binary indicator. The contribution of an approximation PO_1 relatively to another approximation PO_2 is the ratio of non-dominated solutions produced by PO_1 in PO^* , which is the set of Pareto solutions of $PO_1 \cup PO_2$ [18]:

$$Cont(PO_1/PO_2) = \frac{\frac{\|PO\|}{2} + \|W_1\| + \|N_1\|}{\|PO^*\|} \quad (2.9)$$

where PO is the set of solutions in $PO_1 \cap PO_2$, W_1 the set of solutions in PO_1 that dominate some solutions of PO_2 and N_1 the set of non-comparable solutions of PO_1 . This value has to be greater than 0.5 to indicate that PO_1 is better than PO_2 in terms of convergence to the Pareto front.

Spread indicator The spread indicator I_s combines the distribution and cardinality to measure the dispersion of the approximated Pareto set A [18]:

$$I_s = \frac{\sum_{u \in A} |\{u' \in A : \|F(u) - F(u')\| > \sigma\}|}{|A| - 1} \quad (2.10)$$

where $F(u)$ is a fitness function and $\sigma > 0$ a neighborhood parameter. The closer is the measure to 1, the better is the spread of the approximated set A .

Binary ϵ -indicator The binary ϵ -indicator is a convergence-based indicator. It will give the quality of a solution front in comparison with a another set, with regards to all objectives. Let us consider a minimization problem with n positive objectives. An objective vector $f^1 = (z_1^1, z_2^1, \dots, z_n^1)$ is said to ϵ -dominate another objective vector $f^2 = (z_1^2, z_2^2, \dots, z_n^2)$ if $\forall 1 \leq i \leq n : z_i^1 \leq \epsilon \cdot z_i^2$, for a given $\epsilon > 0$. A binary ϵ -indicator $I_\epsilon(A, B)$ gives the factor ϵ such that for any solution in B there is at least one solution in A that is not worse by a factor of ϵ in all objectives. $I_\epsilon(A, B)$ can be calculated as follows [55]:

$$I_\epsilon(A, B) = \max_{z^2 \in B} \min_{z^1 \in A} \max_{1 \leq i \leq n} \frac{z_i^1}{z_i^2} \quad (2.11)$$

Unary hypervolume indicator The hypervolume is an hybrid indicator. Since we already cited a binary indicator (*epsilon*), we will define the hypervolume indicator I_H in its unary form. I_H , associated with an approximation set A is given by the volume of the space portion that is weakly dominated by the set A [18].

2.4.3 Multi-criteria decision aid

Once the Pareto frontier is obtained or approximated, the compromise solutions can be found by establishing a preference model of the decision maker facing several conflicting solutions. Those models can be classified into three broad categories [41, 56] whose methods will be detailed in Section 2.4.5:

1. *Aggregation methods*: numerical scores are calculated by aggregating the criteria to determine the level of preference for a solution. The most known aggregation methods are the Multi-Attribute Utility Theory (MAUT) [57] and the Analytic Hierarchy Process [58].
2. *Interactive methods*: it is a sequential process composed by alternating computation steps and dialogue with the decision maker. A first compromise is submitted to the decision maker who can accept or deny it. If the solution is denied, the DM can give extra information (e.g. releasing a constraint) about his preferences (dialogue) and a new solution can be calculated, so a new decision process begins. Otherwise, no better solution can be found and the process stops. Among the most known interactive methods, the STEP Method (STEM) [59] or the Satisficing Trade-Off Method (STOM) [60] can be cited.
3. *Outranking methods*: the solutions are compared pairwise which enables the possibility to identify the relationship between the solutions. This shows the preference for a solution in comparison to another one. PROMETHEE [61] and ELECTRE [62] are among the most known outranking methods.

Generally, the purpose of MCDA is to provide answers for three main problematic [63]:

1. *The choice problematic (P. α)*: the aid aims the selection of a small number of good solutions in such way that one or several compromise solutions can be chosen.

Example 2.9. *In circuit design, the objective would be to choose the best compromise CPU in terms of performance and price.* ■

2. *The sorting problematic (P. β)*: the aid aims the assignment of each solution to a predefined (ordered) category.

Example 2.10. Depending on performance, price, radiation resistance, thermal operational range, electronic components can be sorted following robustness constraints for commercial, industrial or military and spatial purposes ■

3. The ranking problematic ($P.\gamma$): the aid aims the complete or partial preorder of all the solutions.

Example 2.11. With a preorder for CPUs based on an assessment of their performance, it is possible to associate a price to each processor depending on their ranking. ■

2.4.4 Preference modelling

As mentioned, preference modelling is an important step in decision making problem. It is necessary to understand how a model is built as it will determine how a method will work based on it and thus the outcoming results.

Definitions

Before introducing some important MCDA methods, let us first define some definitions about preference modelling in order to ease the understanding of the following sections.

When modelling the decision maker's preferences, three binary relations which result from the comparison of two alternatives a_i and $a_j \in \mathcal{A}$ are defined [41]:

$$\begin{cases} a_i P a_j & \text{if } a_i \text{ is prefered to } a_j \\ a_i I a_j & \text{if } a_i \text{ is indifferent to } a_j \\ a_i R a_j & \text{if } a_i \text{ is incomparable to } a_j \end{cases} \quad (2.12)$$

These relations express situations of preference, indifference and incomparability and it can be assumed that they satisfy the following properties:

$$\forall a_i, a_j \in \mathcal{A} \begin{cases} a_i P a_j \Rightarrow a_i \neg P a_j : & P \text{ is asymmetric} \\ a_i I a_i & : I \text{ is reflexive} \\ a_i I a_j \Rightarrow a_j I a_i & : I \text{ is symmetric} \\ a_i \neg R a_i & : R \text{ is irreflexive} \\ a_i R a_j \Rightarrow a_j R a_i & : R \text{ is symmetric} \end{cases} \quad (2.13)$$

Intuitively:

- aPb corresponds to the existence of clear and positive reasons that justify significant preference in favour of a
- aIb corresponds to the existence of clear and positive reasons that justify equivalence between the two alternatives

- aRb corresponds to an absence of clear and positive reasons that justify any of the two preceding relations

As stated, (P, I, R) are binary relations. However, these relations can be poor in terms of information about the preferences of a decision maker. Indeed, P will for instance express a strict preference between two alternatives whereas the DM can have a degree of intensity of the preference. For such kinds of situations, the preferences can be defined by a valued preference relation. This valued relation will often take values in the interval $[0; 1]$ [63].

2.4.5 Some important multi-criteria methods

Multi-Attribute Utility Theory

Multi-Attribute Utility Theory (MAUT) has been introduced by Fishburn [64] and Keeney and Raiffa [65]. This method belongs to the family of aggregation methods that consist in substituting the initial multi-criteria problem

$$\min\{f_1(x), f_2(x), \dots, f_m(x) | x \in \mathcal{A}\} \quad (2.14)$$

the following uni-criterion problem:

$$\min\{U(x) | x \in \mathcal{A}\} \quad (2.15)$$

where $U(x)$ is called the utility function that aggregates all the criteria to a single criterion:

$$U(x) = U[f_1(x), f_2(x), \dots, f_m(x)] \quad (2.16)$$

Generally, this utility function is a non-linear function defined such that:

$$U(a) > U(b) \Leftrightarrow a \text{ is preferred to } b \quad (2.17)$$

$$U(a) = U(b) \Leftrightarrow a \text{ is indifferent to } b \quad (2.18)$$

One of the most used utility function is the weighted sum:

$$U(x) = \sum_{j=1}^m w_j f_j(x) \quad (2.19)$$

where w_j is the weight associated to the criterion j .

With this utility function, it is then possible to compute an aggregated score for each solutions and rank them in order to choose among the best ones.

MAUT has been applied in numerous cases and developments have been provided to axiomatize this method and justify its use [57].

Analytical Hierarchy Process (AHP)

Analytical Hierarchy Process (AHP) has been developed by Saaty [58]. This multi-criteria method allows to face structurally complex choices by decomposing the problem in several sub-problems that can be analysed independently and are easier to understand. Similarly to PROMETHEE and ELECTRE, AHP proceeds by making pairwise comparisons of the alternatives, but for example on basis of a ordinal scale from 1 to 9. Indeed, one of the distinctive features of this methods is to build a matrix by asking the decision maker to compare all pairs of alternatives and criteria. Therefore, the input for AHP is not an evaluation table but the DM's preference matrix. The normalized right-hand eigenvector of this matrix is then used to compute the score associated to each alternative and the weight associated to each criterion.

In order to illustrate AHP, we will give more details on a particular case where only the criteria are compared. The decision maker will make pairwise comparisons and give an ordinal scale of preference for the criteria. The following matrix can be obtained:

$$A = \begin{pmatrix} 1 & a_{12} & \dots & a_{1j} & \dots & a_{1m} \\ \frac{1}{a_{12}} & 1 & \dots & a_{2j} & \dots & a_{2m} \\ \vdots & & & & & \\ \frac{1}{a_{1j}} & \frac{1}{a_{2j}} & \dots & a_{ij} & \dots & a_{im} \\ \vdots & & & & & \\ \frac{1}{a_{1m}} & \frac{1}{a_{2m}} & \dots & a_{im} & \dots & 1 \end{pmatrix} \quad (2.20)$$

where a_{ij} is expresses the relative importance of the criterion i over the criterion j .

From this matrix, AHP uses a method based on eigenvector to extract the related weights of each criterion that can be used, for instance, as input data for MAUT in a weighted sum.

A comparison matrix is said to be consistent if $a_{ij}a_{jk} = a_{ik} \forall i, j, k$. However, consistency cannot always be reached and AHP's developers have defined a Consistency Index (CI):

$$CI = \frac{\lambda_{max} - m}{m - 1} \quad (2.21)$$

where λ_{max} is the largest eigenvalue of the matrix and m is the matrix size.

This Consistency Index is then compared to Random (consistency) Index (RI) which are considered to be appropriate CIs. These RIs are obtained by randomly generating matrices and taking the average CI values.

A Consistency Ratio (CR) then is defined:

$$CR = \frac{CI}{RI} \quad (2.22)$$

If the value of the Consistency Ratio is lower or equal to 10%, the inconsistency is considered to be acceptable. Otherwise, the decision maker has to revise judgements.

STEP Method (STEM)

The STEP Method has been proposed by Benayoun [59]. STEM is an interactive and iterative exploration procedure that aims to reach the best compromise according the decision maker after a certain number of cycles. Each cycle is composed of a calculation phase and a decision-making phase (discussion with the decision maker):

1. An efficient compromise solution is determined.
2. This solution is submitted to the decision maker. Three cases can then happen:
 - (a) The decision maker is satisfied and the procedure ends;
 - (b) The decision maker wants to simultaneously improve all the evaluations. This is impossible since the proposed solution is efficient. The procedure ends and cannot help the decision maker.
 - (c) The decision maker identifies a particular criterion on which a concession can be made in order to improve other criteria. A new efficient solution can then be determined.
 - (d) This new solution is submitted. Go to step 2.

Satisficing Trade-Off Method (STOM)

The STOM method has been proposed by Nakayama [60]. Similarly to STEM, it relies on a discussion with the decision maker but is based on the setting of an ideal point defined as follows:

Definition 2.12 (Ideal point). *The ideal point $f^* = (f_1^*, f_2^*, \dots, f_m^*)$ is defined such that $f_i^* = \min\{f_i(x), \forall i = 1, 2, \dots, m, \forall x \in \mathcal{A}\}$.*

The ideal point possesses as coordinates the best values that can be achieved for each criterion separately.

STOM can be summarized in four steps:

1. The first step is to set the ideal point.
2. Then the aspiration level for each criterion is asked to the decision maker; this is the reference point for each criterion of the decision maker. Note that the aspiration level should be lower than the ideal point.
3. A Pareto solution nearest to the ideal point and in the direction of the aspiration level is determined.
4. This solution is submitted to the decision maker. If it is satisfactory, the procedure ends. Otherwise, the decision maker is asked to trade off to define another aspiration level. Go to step 3.

The PROMETHEE methods

PROMETHEE (Preference Ranking Organisation METHod for Enrichment Evaluations) has been initiated by Brans [61] and developed with Mareschal [66] and Vincke [67]. In this section, we will only describe the basics of PROMETHEE. More details can be found in [68].

The PROMETHEE methods are based on the three following steps:

- Enriching the preference structure: a preference function is introduced to characterize a valued relation (see Section 2.4.4)
- Enriching the dominance relation: a valued outranking relation is determined
- Decision aid: the valued outranking relations are exploited

1. *Preference function*

Since the dominance relation is really poor (binary relation), a preference function $P_k(a_i, a_j)$ will be introduced to enrich it. This function gives the preference degree of an alternative a_i over an alternative a_j with respect to the function $d_k(a_i, a_j) = f_k(a_i) - f_k(a_j)$ which is the difference between the evaluation of a_i and a_j for the criterion k , assuming a non decreasing function. Of course, this difference of evaluations has to respect some hypothesis such as having the same interval scale.

Consequently, it is therefore possible to define several types of preference functions based on preference (P) or indifference (Q) thresholds, as shown in Table 2.2. Below the indifference threshold, the decision maker will consider having no preference while above the preference threshold, the decision maker will have a strict preference.

2. *Valued outranking relation*

Multi-criteria preference index

The multi-criteria preference index is defined as follows:

$$\pi(a_i, a_j) = \sum_{k=1}^m P_k(a_i, a_j).w_k, \forall i \neq j \text{ with } \sum_{k=1}^m w_k = 1 \quad (2.23)$$

where $w_k > 0, k = 1, 2, \dots, m$ are the weights on each criterion. $\pi(a_i, a_j)$ represents a measure of the preference of a_i over a_j on all the criteria.

Let us note the following properties of the preference index:

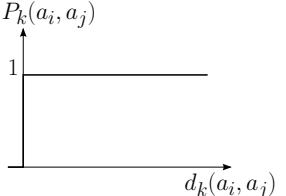
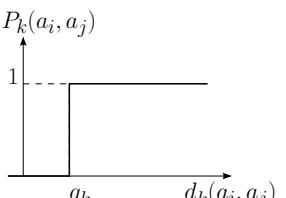
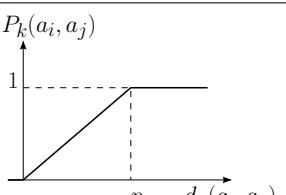
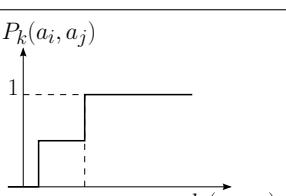
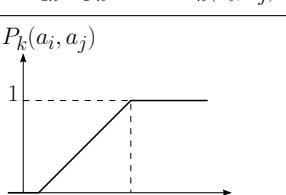
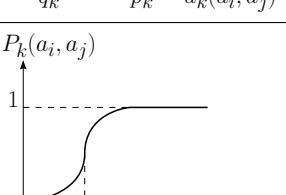
$$\pi(a_i, a_i) = 0 \quad (2.24)$$

$$0 \leq \pi(a_i, a_j) \quad (2.25)$$

$$\pi(a_i, a_j) + \pi(a_j, a_i) \leq 1 \quad (2.26)$$

Outranking flow

Table 2.2 – Preference functions (reproduced from [7])

Usual		Strict preference
U-shape		Q: indifference threshold
V-shape		P: preference threshold
Level		Q: indifference threshold P: preference threshold
Linear		Q: indifference threshold P: preference threshold
Gaussian		S: preference threshold

An “outranking flow” is then defined on the basis of the preference index. That allows to compare alternatives with each others. Three types of flow are formulated:

- The positive outranking flow: $\phi^+ = \frac{1}{n-1} \sum_{j \neq i} \pi(a_i, a_j)$. This flow expresses how a_i outranks all the other alternatives.
- The negative outranking flow: $\phi^- = \frac{1}{n-1} \sum_{j \neq i} \pi(a_j, a_i)$. This flow expresses how a_i is outranked by all the other alternatives.
- The net flow: $\phi(a) = \phi^+(a_i) - \phi^-(a_i)$. This flow expresses the balance between the positive and negative flows of a_i

Let us note the following properties for these flows:

$$\phi^+, \phi^- \in [0; 1] \quad (2.27)$$

$$\phi \in [-1; 1] \quad (2.28)$$

Based on these flows, the PROMETHEE methods will establish an outranking relation.

3. *PROMETHEE I*

The positive and negative flows allow to sort the alternatives of A . Let (S^+, I^+) and (S^-, I^-) be the two complete pre-orders obtained from these flows:

$$\begin{cases} a_i S^+ a_j \Leftrightarrow \phi^+(a_i) > \phi^+(a_j) \\ a_i I^+ a_j \Leftrightarrow \phi^+(a_i) = \phi^+(a_j) \end{cases} \quad (2.29)$$

This means that the higher the positive flow is, the better the alternative.

$$\begin{cases} a_i S^- a_j \Leftrightarrow \phi^-(a_i) < \phi^-(a_j) \\ a_i I^- a_j \Leftrightarrow \phi^-(a_i) = \phi^-(a_j) \end{cases} \quad (2.30)$$

This means that the lower the negative flow is, the better the alternative.

PROMETHEE I establishes a partial ranking by taking the intersection of these two pre-orders:

$$\begin{cases} a_i P^{(1)} a_j \Leftrightarrow \begin{cases} a_i S^+ a_j \text{ and } a_i S^- a_j \\ a_i S^+ a_j \text{ and } a_i I^- a_j \\ a_i I^+ a_j \text{ and } a_i S^- a_j \end{cases} \\ a_i I^{(1)} a_j \Leftrightarrow a_i I^+ a_j \text{ and } a_i I^- a_j \\ a_i R^{(1)} a_j \text{ otherwise} \end{cases} \quad (2.31)$$

where $(P^{(1)}, I^{(1)}, R^{(1)})$ represent respectively the preference, the indifference and the incomparability in PROMETHEE I.

- $a_i P^{(1)} a_j$ (“ a_i is preferred to a_j ”): a_i is simultaneously better and less worse than a_j .
- $a_i I^{(1)} a_j$ (“ a_i and a_j are indifferent”): a_i is neither better nor worse than a_j .
- $a_i R^{(1)} a_j$ (“ a_i and a_j are incomparable”): a_i is better than a_j on some criteria while a_j is better than a_i on other criteria.

4. PROMETHEE II

In order to obtain a complete ranking, the net flow will be considered:

$$\begin{cases} a_i P^{(2)} a_j \Leftrightarrow \phi(a_i) > \phi(a_j) \\ a_i I^{(2)} a_j \Leftrightarrow \phi(a_i) = \phi(a_j) \end{cases} \quad (2.32)$$

where $P^{(2)}$ et $I^{(2)}$ represent respectively the preference and the indifference in PROMETHEE II. This means that the higher the net flow is, the better the alternative.

Let us note that, unlike PROMETHEE I, PROMETHEE II does not give place to incomparability and a complete ranking can directly be obtained.

5. The GAIA plane

When working with more than three criteria, it is impossible to have a perfect visual representation of the solutions space. The GAIA (Geometrical Analysis for Interactive Assistance) plane can give a visualization even if there are more than three criteria, by means of the principal component analysis (PCA) of the net flows on the decision maker's preferences for each criterion.

The PCA allows a projection of the alternatives on a plane that minimizes the loss of information induced by this projection.

This plan allows to have a visual descriptive analysis with several criteria. It can highlight the conflict and synergy between criteria and show the profiles of the alternatives. This will help to identify the potential compromise solutions.

In order to illustrate the use of the GAIA plane, we will take a simple example: six alternatives evaluated on 5 criteria. The associated GAIA plane is given in Figure 2.3.

Four distinctive visual information are shown:

- (a) The green axes that represent the projections of each criterion's axis.
- (b) The blue dots that represent the projection of each solution's uni-criterion net flow. The value of the uni-criterion net flow is read by projecting the point on the related criterion axis.
- (c) The red axis that represents the *decision axis* which is the projection of the set of weights and gives the decision direction.

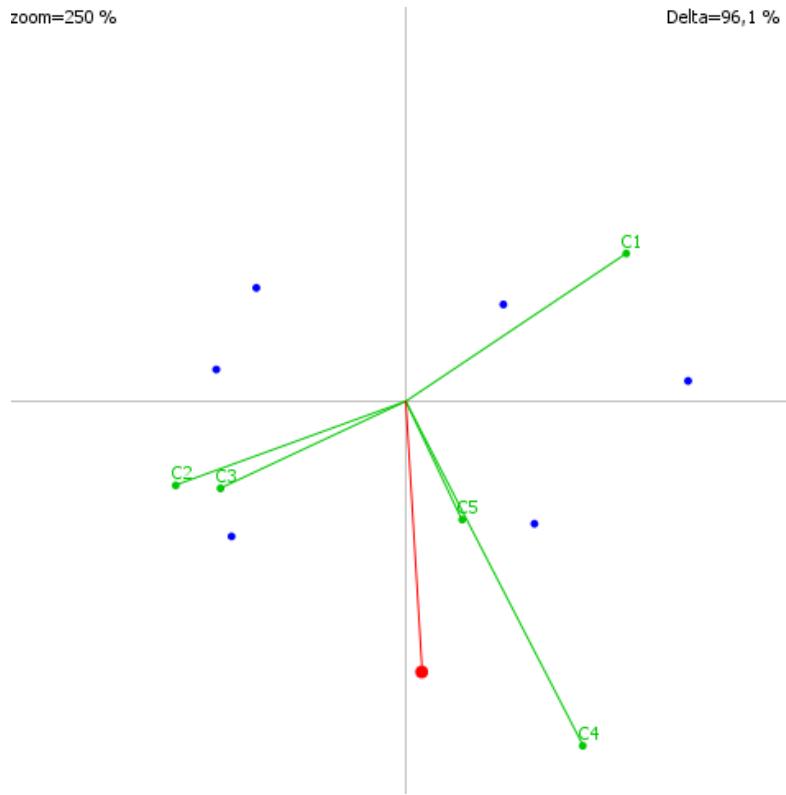


Figure 2.3 – Example of a GAIA plane

- (d) The *delta* value that represents the percentage of kept information since there are projection errors.

From the GAIA plane, we can observe how the criteria are related between each other. Indeed, criteria axes that have opposite directions are conflicting, whereas criteria with the same direction are in synergy. In this case, we can see that C1 is conflicting with C2 and C3, which is to be expected. Also, C4 and C5 have the same direction.

As for the decision axis, it allows a decision maker to know in which direction the best compromise solutions are located based on the criteria weights. Indeed, the alternatives with the highest net flow score will have their furthest projection on that axis, in the direction of that axis. This visually represents the PROMETHEE II ranking, provided that the *delta* value shows that enough information has been kept with the projection.

The ELECTRE methods

ELECTRE (*ELimination Et Choix Traduisant la REalité*, or ELimination and Choice Translating REality) has been developed by Roy [62]. In this section, we will only describe the basics of ELECTRE. More details can be found in [69].

1. ELECTRE I

ELECTRE I is a method linked to the $P.\alpha$ problematic that aims to obtain a subset N of alternatives such that all the solutions that do not belong to this set are outranked by at least one alternative of N and the solutions of N do not outrank each other. N is therefore not the set of good alternatives but rather the set where the best compromise can certainly be found.

The outranking relation is obtained by establishing a weight w_k for each criterion. A concordance index is the associated to each pair (a_i, a_j) of alternatives:

$$c(a_i, a_j) = \frac{1}{W} \sum_{j: f_k(a_i) \leq f_k(a_j)} w_k, \text{ where } W = \sum_{k=1}^m w_k, w_k > 0 \quad (2.33)$$

The concordance index represents a measure of the arguments favourable to the statement “ a_i outranks a_j ”.

A discordance index can also be defined:

$$d(a_i, a_j) = \begin{cases} 0 & \text{if } f_k(a_i) \geq f_k(a_j), \forall k \\ \frac{1}{\delta} \max_k [f_k(a_j) - f_k(a_i)] & \text{otherwise} \end{cases} \quad (2.34)$$

The discordance index is therefore higher if the preference of a_j over a_i is strong on at least one criterion.

Then concordance \hat{c} and discordance \hat{d} thresholds are defined alongside the outranking relation S :

$$\forall i \neq j, a_i S a_j \text{ iff } \begin{cases} c(a_i, a_j) \geq \hat{c} \\ d(a_i, a_j) \leq \hat{d} \end{cases} \quad (2.35)$$

From this definition, a subset N of alternatives (called the kernel) is established such that:

$$\begin{cases} \forall a_j \in A \setminus N, \exists a_i \in N : a_i S a_j \\ \forall a_i, a_j \in N, a_i \overline{S} a_j \end{cases} \quad (2.36)$$

A subset N of alternatives is established such that all the alternatives that do not belong to this set is outranked by at least one alternative of N and the alternatives of N are incomparable. The decision process will therefore take place within the set N . The kernel exists and is unique when the outranking relation S does not contain circuit in its graph.

2. ELECTRE II

This method aims to rank the alternatives. The outranking relation is defined by fixing two concordance thresholds \hat{c}_1 and \hat{c}_2 such that $\hat{c}_1 > \hat{c}_2$ and by building a strong outranking relation S^F and a weak outranking relation S^f based on these two thresholds:

$$a_i S^F a_j \text{ iff } \begin{cases} c(a_i, a_j) \geq \hat{c}_1 \\ \sum_{k: f_k(a_i) > f_k(a_j)} w_k > \sum_{k: g_k(a) < g_k(b)} w_k \\ (f_k(a_i), f_k(a_j)) \notin D_k, \forall k \end{cases} \quad (2.37)$$

$$a_i S^f a_j \text{ iff } \begin{cases} c(a_i, a_j) \geq \hat{c}_2 \\ \sum_{k: f_k(a_i) > f_k(a_j)} w_k > \sum_{k: f_k(a_i) < f_k(a_j)} w_k \\ (f_k(a_i), f_k(a_j)) \notin D_k, \forall k \end{cases} \quad (2.38)$$

The discordance can also induce two levels of relations by building two sets of discordance for each criterion.

In order to obtain the ranking, a set is determined from S^F . This set B contains the alternatives that are not strongly outranked by any others. From B and S^f , the set A^1 of alternatives that are not weakly outranked by any alternatives of B is determined. The set A^1 constitutes the best alternatives class. A^1 is then removed and the process is repeated to find A^2 and so on until a complete pre-order is obtained.

Let us note that a second complete pre-order can be obtained by applying the process first with the less good alternatives class and then the best ones.

3. ELECTRE III

This method takes into account the indifference and preference thresholds. It is based on a valued outranking relation that is less sensible to data and parameters variabilities.

In ELECTRE III, an outranking degree $S(a_i, a_j)$ associated to each pair (a_i, a_j) of alternatives is defined. It can be understood as an “degree of credibility of outranking” of a_i over a_j .

A weight w_k is associated to each criterion and for each pair (a_i, a_j) of alternatives the concordance index is computed as follows:

$$c(a_i, a_j) = \frac{1}{W} \sum_{k=1}^m w_k c_k(a_i, a_j), \text{ where } W = \sum_{k=1}^m w_k \quad (2.39)$$

with

$$c_k(a_i, a_j) = \begin{cases} 1 & \text{if } f_k(a_i) + q_k(f_k(a_i)) \geq f_k(a_j) \\ 0 & \text{if } f_k(a_i) + p_k(f_k(a_i)) \leq f_k(a_j) \\ \text{linear} & \text{if } \begin{aligned} f_k(a_i) + q_k(f_k(a_i)) &\leq f_k(a_j) \\ &\leq f_k(a_i) + p_k(f_k(a_i)) \end{aligned} \end{cases} \quad (2.40)$$

where q_k et p_k represent respectively the indifference and preference thresholds.

The definition of discordance is then enriched by the introduction of a veto threshold $v_k(f_k(a_i))$ for each criterion k such that any credibility for the outranking of a_j by a_i is refused if $f_k(a_j) \geq f_k(a_i) + v_k(f_k(a_i))$.

A discordance index is then defined:

$$D_k(a_i, a_j) = \begin{cases} 0 & \text{if } f_k(a_j) \leq f_k(a_i) + p_k(f_k(a_i)) \\ 1 & \text{if } f_k(a_j) \geq f_k(a_i) + v_k(f_k(a_i)) \\ \text{linear} & \text{if } \begin{aligned} f_k(a_i) + p_k(f_k(a_i)) &\leq f_k(a_j) \\ &\leq f_k(a_i) + v_k(f_k(a_i)) \end{aligned} \end{cases} \quad (2.41)$$

The degree of outranking is finally defined:

$$S(a_i, a_j) = \begin{cases} c(a_i, a_j) & \text{if } D_k(a_i, a_j) \leq c(a_i, a_j) \\ c(a_i, a_j) \prod_{k \in \mathcal{F}(a_i, a_j)} \frac{1 - D_k(a_i, a_j)}{1 - c(a_i, a_j)} & \text{otherwise} \end{cases} \quad (2.42)$$

where $\mathcal{F}(a_i, a_j)$ is the set of criteria for which $D_k(a_i, a_j) > c(a_i, a_j)$. The degree of outranking is thus equal to the concordance index when no criterion is discordant, otherwise the concordance index is decreased proportionally depending on the importance of the discordances.

A value $\lambda = \max_{a_i, a_j \in \mathcal{A}, i \neq j} S(a_i, a_j)$ is determined and only the outranking degree that have a value greater or equal to $\lambda - s(\lambda)$, where $s(\lambda)$ is a threshold to be determined, are considered. A ranking can then be determined from a qualification index $Q(a)$ for each alternative a that represents the difference between the number of outranked alternatives by a and the number of alternatives that outrank a . The set of actions having the largest qualification will be called the first distillate D_1 .

If D_1 contains only one alternative, the previous procedure is repeated with $\mathcal{A} \setminus D_1$. Otherwise the same procedure is applied for D_1 and if the obtained distillate D_2 contains only one alternative, the procedure is repeated with $D_1 \setminus D_2$. Otherwise, it is applied for D_2 , and so on until D_1 is completely

used, before starting with $A \setminus D_1$. This procedure produces a first complete preorder.

A second complete preorder can be obtained by applying the opposite procedure where the alternatives with the smallest qualification are first used.

2.5 Conclusion

In this chapter, we have given a short overview of multi-objective optimization and multi-criteria decision aid. We have explained in what kind of context a multi-criteria paradigm can be applied and have presented some of the classical method used in the field.

In the next chapter, we will define the problem we tackle and show how a 3D-SIC can be modelled in order to apply multi-objective optimization.

3

Problem definition, modelling and simulation results

Chapter abstract

In this chapter, we define the design problem we tackle: the 3D partitioning with floorplanning estimation. We then present the criteria we consider and show how a 3D-SIC can be modelled in order to apply multi-objective optimization. Simulations are ran based on an existing platform and the obtained results can provide qualitative and quantitative information to a designer such as trade-off analysis or what would be the advantages of using 3D-SICs with respect to some criteria, and those analyses would not be available with current tools. Then, since the first case study was limited, the methodology is tested with a more realistic case study that contains more functional components, to show that a multi-criteria paradigm does give added value compared to a uni-criterion approach, in terms of design space analysis. Finally, the robustness of the algorithm is studied in order to show that it can have good convergence and diversity properties despite the complexity of the problem.

Associated publications:

- N.A.V. Doan, F. Robert, Y. De Smet, D. Milojevic, "MCDA-based methodology for efficient 3D-design space exploration and decision", *International Symposium on System-on-Chip Proceedings (SOC 2010)*, Tampere (Finland), pp. 76-83, September 2010
- N.A.V. Doan, Y. De Smet, F. Robert, D. Milojevic, "On the use of multi-criteria decision aid tools for the efficient design of 3D-stacked integrated

- circuits: a preliminary study", *Proceedings of IEEE International Conference on Industrial Engineering and Engineering Management (IEEM 2010)*, Macau (China), December 2010
- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "A MOO-based methodology for designing 3D-stacked integrated circuits", *Journal of Multi-Criteria Decision Analysis*, vol.21, no. 1-2, pp. 43-63, January-April 2014

3.1 Introduction

In Chapter 1, we have presented a review of the literature about the field of microelectronics design. We have highlighted some limitations of the current tools that already occur for 2D-ICs. In this chapter, we will define the 3D partitioning problem, which is the issue we tackle and show how we model it in order to propose improvements to design flows.

3.2 Problem definition

As stated in Chapter 1, the limitations of the current design flows can be summarized in three points:

- Limitation of the design space exploration due to huge runtime
- Unicriterion optimization or limited trade-off analysis on a limited number of criteria
- Few 3D-SIC dedicated tools

In order to address these limitations, we propose in this thesis a methodology based on multi-objective/criteria tools and taking into account 3D-SIC specificities to explore the design space.

While this methodology could be applied at different levels in a design flow, we have focused our development in the logical design step and the virtual prototyping flow, more specifically the partitioning with floorplanning estimation and performance assessments.

3.2.1 Designing an IC

In order to meet the specifications, a design has first to make a choice at a physical level:

- Targeted architecture, e.g. ASIC, FPGA
- Number of functional units
- Number of memories and their size
- The general layout
- ...

Since the 3D-SICs are based on conventional circuits, the options and degrees of freedom coming from 2D-ICs are still present:

- Process technology, e.g. 180 nm to 22 nm CMOS
- Memories technology, e.g. SRAM, DRAM, FLASH
- Communication infrastructure, e.g. bus, Network-on-Chip
- ...

In addition to those options and degrees of freedom coming from 2D-ICs, numerous 3D-SIC's parameters appear [4]:

- Number of tiers to use
- Place and route of the functional units between the tiers
- Technology to use per tiers (heterogeneity)
- ...

The above mentioned parameters illustrate the numerous possibilities for designing a circuit and how the design space for 2D-ICs becomes much larger when considering 3D-SICs. The main issue is therefore to choose the most efficient combination among all those options. This can thus be compared to a combinatorial optimization problem. Also, given the multi-criteria nature of designing 3D-SIC, we choose to take into account all the criteria simultaneously for the optimization. In our case, due to the heterogeneous nature of the criteria (see Section 3.3), we have few hopes to successfully adopt an exact method and we will therefore use metaheuristics which are commonly-used tools for such kinds of problems. In the next section will define the criteria that a designer can consider.

3.3 Model and criteria definition

Typically, the criteria that have to be optimized simultaneously can be the performance, the power consumption, the cost, the package size, the heat dissipation, etc. In this model, we will decide to consider five criteria which are among the most important parameters while designing a circuit [38]:

1. *The total interconnection length:* this parameter can reflect the global performance of a system. The objective is to minimize it in order to have, for instance, a short delay and low power consumption. It will be calculated using the Manhattan distance [70]:

$$d_{i,j} = |x_i - x_j| + |y_i - y_j| \quad (3.1)$$

where (x_k, y_k) is the geometrical coordinates of the k^{th} block. As a first approximation, the center point of each block will be selected as reference coordinate. Also, since it is more interesting to place close to each other two blocks that require a large bandwidth (BW) to communicate, we will balance

the values as follows:

$$d'_{i,j} = \frac{d_{i,j}}{BW_{ij}} \quad (3.2)$$

where BW_{ij} is the bandwidth required between the block i and the block j . The global interconnection length D will be the sum of $d'_{i,j}$ for all communicating blocks:

$$D = \sum d'_{i,j} \quad (3.3)$$

2. *The cost*: an economical factor is obviously an important criteria for a design. This criteria has been estimated with the aid of an expert in 3D-SIC manufacturing. While a circuit can be more efficient with many layers, it will also be more expensive. This criteria has to be minimized. Due to the confidential nature of the cost of a 3D-SIC, we will consider a simplified model where the cost is proportional to the area and increasing exponentially with the number of tiers:

$$cost = a(tech).S + b(tech)^{\text{layer number}} \quad (3.4)$$

where $a(tech)$ and $b(tech)$ are coefficient depending on the technology assigned. Let us note that this criterion includes both discrete and continuous variables.

3. *The package volume*: this can be an important criteria when designing embedded circuits. The package volume is calculated as follows:

$$volume = \text{largest layer size} * \text{stack thickness} * \text{number of tiers} \quad (3.5)$$

A large approximation of $20 \mu\text{m}$ will be made for the thickness of one tier. Let us note that this criterion includes both discrete (number of tiers) and continuous variables (layer size).

4. *The clock source position*: in this model, we consider a synchronous system so the objective is to minimize the distance between each block and the clock source in order to have a high frequency. We choose arbitrarily to approximate the reference point as a fixed point located at the upper left corner of the middle tier of the 3D-SIC.
5. *The thermal dissipation*: thermal dissipation is one of the major issues when designing 3D-SICs. It can be more appropriate to place two blocks underneath each other in successive tiers but a high heat dissipation may happen in intensive computational process. This criterion is a research topic on its own [23, 71]. Here we will use a simplified evaluation model with finite elements. This model will consider that the dissipated power, intra- or inter-tiers, is inversely proportional to the distance to the heat source:

$$P_{diss} = P_{comp} \sum_{i=1}^n \frac{1}{R_{th,i} \cdot r} \quad (3.6)$$

where r the distance to the heat source and $R_{th,i}$ the thermal resistance depending on whether the dissipation is intra- or inter- tiers. This criterion is still on early development stage in our research and we can generate thermal maps of a partition as shown in Figure 3.1 but this is currently based on finite elements [72] which require quite a long computational time even for a simplified thermal model. This criterion in its current development stage is difficult to integrate to the exploration process, due to the computation time of finite elements methods. In the current work, we will simply compute the peak power of a circuit which can be done more quickly.

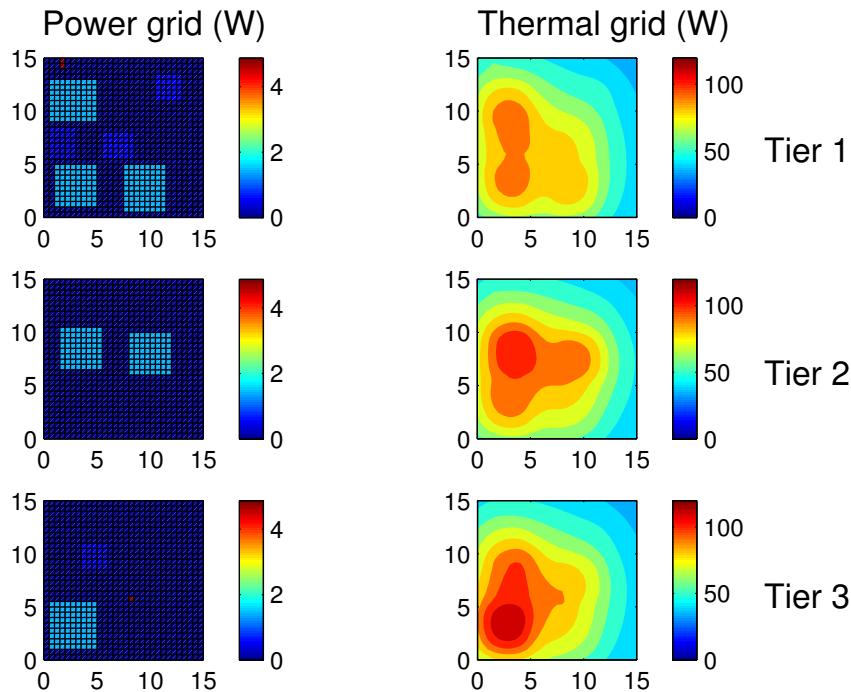


Figure 3.1 – Power grid and thermal map of a partition (3 tiers)

A recap table containing the criteria is shown in Table 3.1.

We will first focus on the three first criteria in order to be able to have a visualization of the design space. We will also arbitrarily introduce some limitations in term of degrees of freedom to analyse what happens if we release a constraint. This will be done while considering the three same criteria, in order to keep a visualization and show how the flexibility of MOO will improve the information and the results. These analyses will be done with a first limited case study to show the applicability of the multi-criteria paradigm

Table 3.1 – Recap table of the criteria

Criterion	Unit	Type
Interconnection length	mm or μm (bandwidth-weighted)	Continuous
Cost	cost unit	Mixed
Volume	mm^3 or μm^3	Mixed
Clock source position	mm or μm	Continuous
Thermal dissipation	W	Continuous

Then will analyse our methodology with the five criteria that have been presented with a second case study which is more realistic in term of functional components and show the added value of a multi-criteria analysis.

3.4 Multi-objective optimization

3.4.1 Overview of the method

In summary, the problem we are facing is to place several blocks that have to be assigned in many tiers while considering multiple conflicting criteria. Now that the criteria have been defined, we will present a proposition of a new design methodology based on multi-objective optimization, with the related model.

As explained in Chapter 1, designing ICs implies numerous choices and designers are likely to freeze a certain amount of choices on basis of their experience. This will therefore limit the exploration of the design space and good solutions may be ignored.

In order to enable a design space exploration, we propose a method in four steps based on MCDA which is illustrated in Figure 3.2. The implementation will be briefly presented in the next section.

For the problem we consider, the input data will contain the information about the scenario:

- Type and number of blocks: computational units, memories, etc.
- Size of the blocks: inherent to the block.
- Minimum aspect ratio: we consider a degree of freedom where a block can have its dimensions varying within a given aspect ratio range. This means that a block does not have to be square, as shown in Figure 3.3. This parameter can influence the delay in a block.
- Size variability of a block: we add this degree of freedom considering that the specified size of a block can be fixed by the designer but this fixed size can restrict the design space exploration. The variability of a block's size can impact the performances and the global footprint.

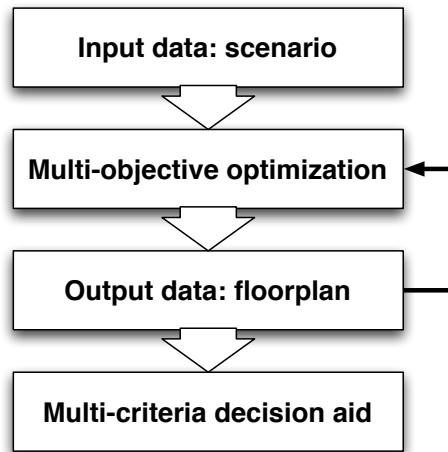


Figure 3.2 – MCDA-based design methodology for 3D DSE partitioning

In addition, the bandwidth requirements are needed as they will indicate which are the important interconnections and prevent two blocks that require a large bandwidth from being too far from each other. The available manufacturing technologies are also useful to enable the design of heterogeneous systems.

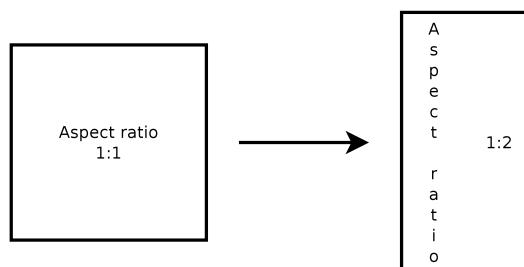


Figure 3.3 – Example of aspect ratio degree of freedom

The combination of all the parameters described in the model are the possible alternatives for a 3D-SIC design and will provide output data after design space exploration. For a partition problem the required output data are generally the geometrical layout of the circuit [38]:

- The geometrical coordinates for each block and the assigned layer.
- The size of each block (if it can vary from the specified size).
- The aspect ratio for each block.
- The technology assigned to each tier: a thinner technology will reduce the

size of each block. The size of a block will define the number of transistors inside using a given technology, for example 180 nm. For a constant number of transistors, if the block is manufactured with a smaller technology, let us say 45 nm, then its size will be divided by a $(180/45)^2$ factor, as shown in Figure 3.4. Please note, that this factor is a rough approximation which is not always met with real physical design and of course this accuracy can be improved.

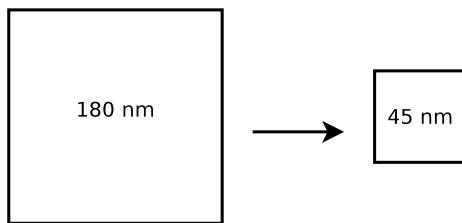


Figure 3.4 – The use of different manufacturing technologies results in a size variation

3.4.2 Implementation of the method

In this section, we will briefly explain the implementation of our design method and show some experimental results based on this first approach. Due to the huge size of the design space, an explicit enumeration of the possible solutions will take considerable time. For instance, for a simplified problem of 3 tiers of 10x10 mm and 5 blocks of 2x2 mm to place, there are 75 possible positions for each block. The number of combinations is given by $\frac{n!}{k!(n-k)!}$, where n is the number of possible positions and k the number of blocks. For this simplified problem, there are therefore 17 259 390 combinations. For 10 blocks to place, the number of possibilities increases to more than 8.10^{11} . Besides, this small calculation does not take into account the numerous other possible choices.

We will therefore apply a metaheuristic and more specifically a NSGA-II algorithm. We chose to use NSGA-II as it is quick to implement which is convenient as proof of concept for our research purposes (show the applicability of MOO/MCDA for 3D-SIC design). In addition, this algorithm is flexible and is known for handling mixed variables since they are only involved for evaluations of the alternatives in the selection step. For the first case study we consider a circuit based on the 3MF MP-SoC platform developed at IMEC [6]. This case study has been implemented using Matlab and all data were coded using matrices.

As shown in Table 3.2, we choose to encode our data in real or integer values, so that they can be used directly by design tools:

- The component identification number (ID) is a fixed integer value linked to the component.
- The assigned layer (L) is a discrete value ranging from 1 to 5 in the case study.
- The geometrical coordinates (X,Y) are real values that depends on the dimension of the circuits and the aspect ratio of a block, so that the component cannot be placed outside the chip.
- The size (S) is a fixed real value linked to the component.
- The aspect ratio (AR) is a real value ranging from AR_{min} to $1/AR_{min}$ where AR_{min} is given as a specification as explained in Section 4.
- The length in X and Y axis (LX, LY) are real values computed from the size and the aspect ratio.
- The assigned technology per layer is a discrete value taking one of the specified technology (see Table 3.12).

This matrix will be our full chromosome for the NSGA-II algorithm (see example in Table 3.2).

Table 3.2 – Output matrix template

ID	L	X	Y	S	AR	LX	LY	T
1	2	4.5	6	18.6	1	4.3128	4.3128	90
2	2	4	0.4	18.6	1	4.3128	4.3128	90
3	3	3.1	6.9	18.6	1	4.3128	4.3128	90
4	3	8.4	10.1	18.6	1	4.3128	4.3128	90
5	3	6.6	2.2	18.6	1	4.3128	4.3128	90
6	1	9	5.7	18.6	1	4.3128	4.3128	90
7	1	10	3.5	0.54	1	0.7348	0.7348	90
8	1	7.5	11	6.74	1	2.5962	2.5962	90
9	2	9	5	6.74	1	2.5962	2.5962	90
10	1	4.5	8	6.62	1	2.5729	2.5729	90
11	2	8.6	0.4	6.62	1	2.5729	2.5729	90
12	3	8.3	7.4	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

We implemented our design space exploration following the steps of the NSGA-II which can be summarized by the diagram shown in Figure 3.5. This implementation does not include an archive since through hundreds of test simulations, an alternative has never been explored twice for the considered case studies.

Initialization (the initial population)

We will work with a minimum size of population, namely 50, which is a common value in GAs [73]. The initial population will be a set of at least 50 solutions with the

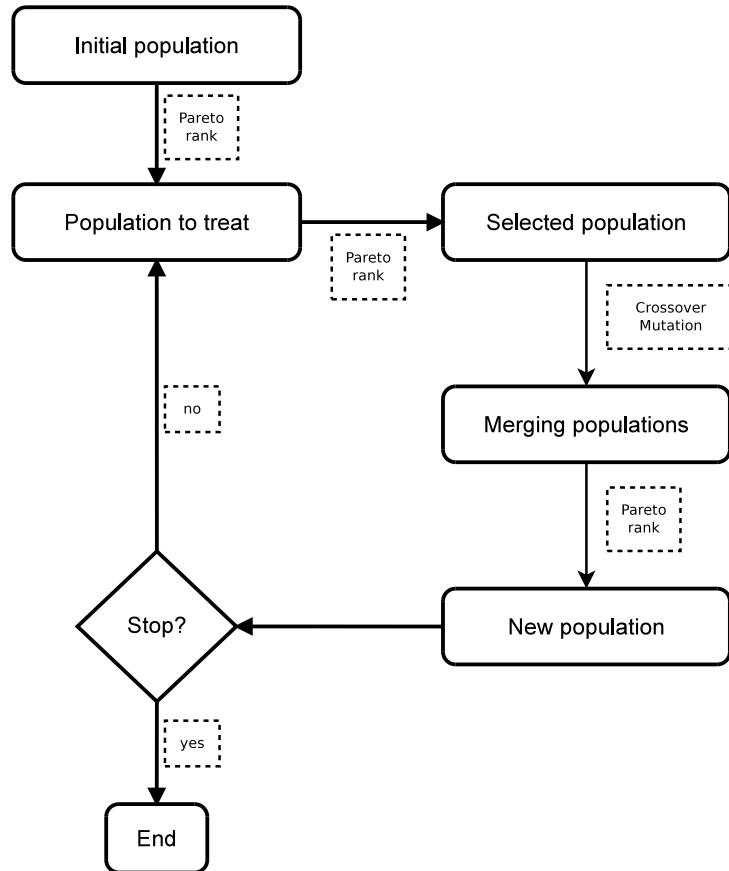


Figure 3.5 – General NSGA-II steps

best Pareto ranks from a randomly-generated set of 10 000 solutions. The produced set places the blocks randomly (using a uniform distribution) and does not allow overlapping between the blocks. These incorrect solutions are simply removed and it is difficult to estimate this proportion as it depends on the problem size (number of blocks). We could of course use a greedy algorithm as well as a more advanced method such as GRASP (Greedy Randomized Adaptive Search Procedure) [74]. This can be done as future work for comparison purposes.

Of course, having at least 50 Pareto solutions does not always happen. Actually, the selection is based on the Pareto rank so it does not include only the Pareto solutions (rank 1), but also the solutions with lower ranks until there are enough solutions.

Selection for crossover

For the selection step, two solutions will be allowed to make a crossover depending on a roulette wheel where the probability is proportional to the normalized Euclidean distance between the solutions ordered by their Pareto rank in the objective space. The normalization is done as follows :

$$\frac{g_i(a_j)}{\max_{a_j \in A} g_i(a_j)} \quad (3.7)$$

where A is the set of alternatives in the Pareto front with $a_j \in A$ and $g_i(a_j)$ is the evaluation of the alternative a_j on the criterion i .

The probability for two solutions to do a crossover will vary linearly with the Euclidean distance between them, as shown in Figure 3.6. The distance between the two furthest alternatives ($d_{furthest}$) will be associated with a probability $P_{c,min}$ while the distance between the two closest alternatives ($d_{closest}$) will be associated with a probability $P_{c,max}$. If two solutions are close to each other, they will have more chance to reproduce than if they are distant. This is to ensure the intensification properties of our algorithm. Therefore, we will have to specify a lower bound ($P_{c,min}$) and an upper bound ($P_{c,max}$) for the crossover probability. $P_{c,min}$ is set for the solutions which are the furthest to each other while $P_{c,max}$ is set for those which are the closest. In between, the probability will vary linearly inside these bounds.

These values will be fixed as [$P_{c,min} = 0.6$; $P_{c,max} = 1.0$] since these seem to be common values [73] and we have empirically observed that intensification properties are ensured.

Crossover

Let us now see how does the crossover occur. First, let us remark that it does not have limitations for the exploration process since the information contained in the matrix spans the whole circuit.

Second, we have to analyse how the chromosome is coded in order to see how we will apply the crossover step. For instance, let us choose the Layer (L) column as indicator for the crossover. If we order the matrix in Table 3.2 following this column, we will have the Table 3.3 and the Table 3.4 for another solution that we will use for the crossover.

Now, without loss of generality, let us suppose that the crossover happens (randomly) on line 7. One of the child will be the Table 3.5 and we see that the original scenario is not preserved since the first column (in bold) contains the same ID several times.

We observe that the only possible indicator for the crossover step is the ID column. Indeed if we order the two parents following the ID column, we have the Tables

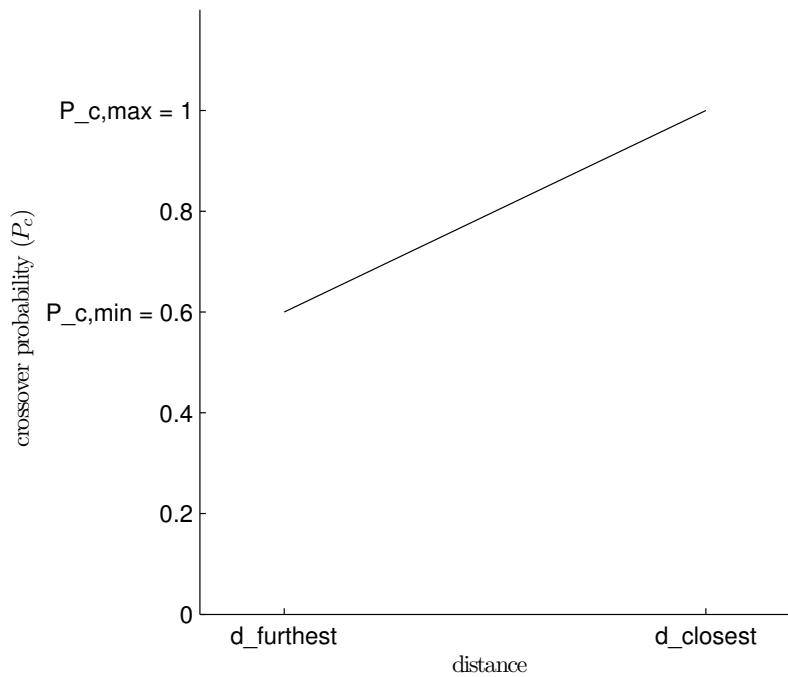


Figure 3.6 – Evolution of the crossover probability as a function of the distance between two solutions

3.6 and 3.7. If we still consider that the crossover occurs on line 7, we can have the child shown in Table 3.8. We see that there is no inconsistency since the scenario is still respected.

Mutation

A mutation cannot happen anywhere in the matrix. Indeed, if we take the conclusion about the choice of the crossover row indicator, all the elements except the ID column can mutate.

The mutation used is a random uniform distribution $U([a, b])$, where $[a, b]$ is the interval of values allowed for the mutation. For the discrete values, we use equidistributed probabilities. The mutation probability of a child will be set as $P_m = 0.3$. Empirical observations have shown that smaller mutation probability can easily lead to a local optimum. This can be explained by the fact that we choose that only one

Table 3.3 – First parent, ordered by L column; the line specifies the crossover cut

ID	L	X	Y	S	AR	LX	LY	T
6	1	9	5.7	18.6	1	4.3128	4.3128	90
7	1	10	3.5	0.54	1	0.7348	0.7348	90
8	1	7.5	11	6.74	1	2.5962	2.5962	90
10	1	4.5	8	6.62	1	2.5729	2.5729	90
1	2	4.5	6	18.6	1	4.3128	4.3128	90
2	2	4	0.4	18.6	1	4.3128	4.3128	90
9	2	9	5	6.74	1	2.5962	2.5962	90
11	2	8.6	0.4	6.62	1	2.5729	2.5729	90
3	3	3.1	6.9	18.6	1	4.3128	4.3128	90
4	3	8.4	10.1	18.6	1	4.3128	4.3128	90
5	3	6.6	2.2	18.6	1	4.3128	4.3128	90
12	3	8.3	7.4	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

Table 3.4 – Second parent, ordered by L column; the line specifies the crossover cut

ID	L	X	Y	S	AR	LX	LY	T
4	1	1.6	8.5	18.6	1	4.3128	4.3128	90
5	1	1.2	1.3	18.6	1	4.3128	4.3128	90
6	1	0.6	4.7	18.6	1	4.3128	4.3128	90
3	2	5.9	4	18.6	1	4.3128	4.3128	90
9	2	5.4	8	6.74	1	2.5962	2.5962	90
10	2	8.5	8.1	6.62	1	2.5729	2.5729	90
11	2	2.8	4.6	6.62	1	2.5729	2.5729	90
1	3	7	6.3	18.6	1	4.3128	4.3128	90
2	3	7.4	9.8	18.6	1	4.3128	4.3128	90
7	3	5.6	5.5	0.54	1	0.7348	0.7348	90
8	3	2.8	5.5	6.74	1	2.5962	2.5962	90
12	3	5.7	7.5	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

single element of a line can mutate instead of the whole line. If a child is forced to mutate, then one randomly-chosen value of the whole matrix will mutate within the range of values it is allowed to take.

A Gaussian mutation is also a common operator but it has not been chosen since it will produce a solution which is not far from the original one. This is not really interesting to have similar solutions when exploring the design space for integrated

Table 3.5 – Possible child, ordered by L column

ID	L	X	Y	S	AR	LX	LY	T
6	1	9	5.7	18.6	1	4.3128	4.3128	90
7	1	10	3.5	0.54	1	0.7348	0.7348	90
8	1	7.5	11	6.74	1	2.5962	2.5962	90
10	1	4.5	8	6.62	1	2.5729	2.5729	90
1	2	4.5	6	18.6	1	4.3128	4.3128	90
2	2	4	0.4	18.6	1	4.3128	4.3128	90
9	2	9	5	6.74	1	2.5962	2.5962	90
1	3	7	6.3	18.6	1	4.3128	4.3128	90
2	3	7.4	9.8	18.6	1	4.3128	4.3128	90
7	3	5.6	5.5	0.54	1	0.7348	0.7348	90
8	3	2.8	5.5	6.74	1	2.5962	2.5962	90
12	3	5.7	7.5	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

Table 3.6 – First parent, ordered by ID column; the line specifies the crossover cut

ID	L	X	Y	S	AR	LX	LY	T
1	2	4.5	6	18.6	1	4.3128	4.3128	90
2	2	4	0.4	18.6	1	4.3128	4.3128	90
3	3	3.1	6.9	18.6	1	4.3128	4.3128	90
4	3	8.4	10.1	18.6	1	4.3128	4.3128	90
5	3	6.6	2.2	18.6	1	4.3128	4.3128	90
6	1	9	5.7	18.6	1	4.3128	4.3128	90
7	1	10	3.5	0.54	1	0.7348	0.7348	90
8	1	7.5	11	6.74	1	2.5962	2.5962	90
9	2	9	5	6.74	1	2.5962	2.5962	90
10	1	4.5	8	6.62	1	2.5729	2.5729	90
11	2	8.6	0.4	6.62	1	2.5729	2.5729	90
12	3	8.3	7.4	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

circuits. Of course, a large standard deviation value can be chosen but this will be likely to produce solution which are out of the feasible bounds.

Consistency test

Of course, infeasible solutions (overlapping blocks or blocks outside of the tier's surface) may appear after the crossover/mutation step, since these operations are

Table 3.7 – Second parent, ordered by ID column; the line specifies the crossover cut

ID	L	X	Y	S	AR	LX	LY	T
1	3	7	6.3	18.6	1	4.3128	4.3128	90
2	3	7.4	9.8	18.6	1	4.3128	4.3128	90
3	2	5.9	4	18.6	1	4.3128	4.3128	90
4	1	1.6	8.5	18.6	1	4.3128	4.3128	90
5	1	1.2	1.3	18.6	1	4.3128	4.3128	90
6	1	0.6	4.7	18.6	1	4.3128	4.3128	90
7	3	5.6	5.5	0.54	1	0.7348	0.73485	90
8	3	2.8	5.5	6.74	1	2.5962	2.5962	90
9	2	5.4	8	6.74	1	2.5962	2.5962	90
10	2	8.5	8.1	6.62	1	2.5729	2.5729	90
11	2	2.8	4.6	6.62	1	2.5729	2.5729	90
12	3	5.7	7.5	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

Table 3.8 – Possible child, ordered by ID column

ID	L	X	Y	S	AR	LX	LY	T
1	2	4.5	6	18.6	1	4.3128	4.3128	90
2	2	4	0.4	18.6	1	4.3128	4.3128	90
3	3	3.1	6.9	18.6	1	4.3128	4.3128	90
4	3	8.4	10.1	18.6	1	4.3128	4.3128	90
5	3	6.6	2.2	18.6	1	4.3128	4.3128	90
6	1	9	5.7	18.6	1	4.3128	4.3128	90
7	1	10	3.5	0.54	1	0.7348	0.7348	90
8	3	2.8	5.5	6.74	1	2.5962	2.5962	90
9	2	5.4	8	6.74	1	2.5962	2.5962	90
10	2	8.5	8.1	6.62	1	2.5729	2.5729	90
11	2	2.8	4.6	6.62	1	2.5729	2.5729	90
12	3	5.7	7.5	0.66	1	0.8124	0.8124	90

ID: component identification number; L: assigned layer;
(X, Y): geometrical coordinate; S: size (mm^2); AR: aspect ratio;
(LX, LY): length in X and Y axis; T: assigned technology for the layer

made with randomness. In order to verify that, we perform a test on each new solution to check if there is overlapping between the blocks. Currently, the solutions which are infeasible will be discarded. Of course, it is possible to apply some repair mechanism, for instance by moving the badly-placed blocks. This is still to be investigated as future work since moving these blocks is not trivial as this can create overlapping elsewhere on a layer.

Termination

Three stop conditions have been implemented and are based on what is commonly used:

- Maximum number of iterations, set to 100.
- Maximum elapsed time, set to 60 minutes.
- Maximum number of iterations with an unchanged population, set to 10.

The maximum elapsed time has been chosen arbitrarily for quick testing purposes. As illustrated in Section 3.4.2, the design space is huge and finding an accurate Pareto frontier can be time consuming. On other hand, NSGA-II has shown that it can quickly produce good approximations [5]. Having a simulation time of a few hours is therefore enough, considering that, in practice, the optimization of one single architecture can take from several days to several weeks with the current design tools. Also, due to the approximation in the model, trying to find a really accurate Pareto front would not have real added value either.

3.5 Experimental set-up and results

3.5.1 Case study 1: basic MPSoc analysis

The 3MF MPSoC is made of 13 blocks as shown on Figure 3.7:

- 6 ADRES processors ([75])
- 2 data memories (L2D#)
- 2 instruction memories (L2Is#)
- 1 external memory interface (EMIF)
- 1 input/output processor (FIFO)
- 1 ARM processor (ARM)

Details about the area required for each component is given in Table 3.9 for a 90 nm technology. This table is also the input matrix required to specify the scenario.

The 3MF MPSoC can be configured for three use cases which have specific bandwidth requirements. For the following results, we will base our simulation on the "data split scenario" configuration which possesses the communication specifications shown in Table 3.10. This information is implemented, as shown in Table 3.11, in an input matrix which is built by specifying the communication structure: the first column will contain the ID of the source block and each next pairs of columns will contain the ID of the target blocks and the bandwidth required.

The input data are thus shown in Tables 3.9 and 3.11. The available technologies are also needed to take advantage of the heterogeneity. An example matrix for this input data is given in Table 3.12. Since no information is given about the ARM unit bandwidth usage, we will simplify our problem and not include it in the implementation. We consider therefore 12 blocks to assign.

Table 3.9 – Scenario input matrix example

Component	ID	Size (90 nm)	Min aspect ratio	Size variability
ADRES 1~6	1~6	18.6 mm ²	0.5	±20%
FIFO	7	0.54 mm ²	0.5	0
L2D1-2	8-9	6.74 mm ²	0.5	+30%
L2Is1-2	10-11	6.62 mm ²	0.5	+30%
EMIF	12	0.66 mm ²	0.5	0.1
ARM	13	0.89 mm ²	0.5	0

ID: Component identification number

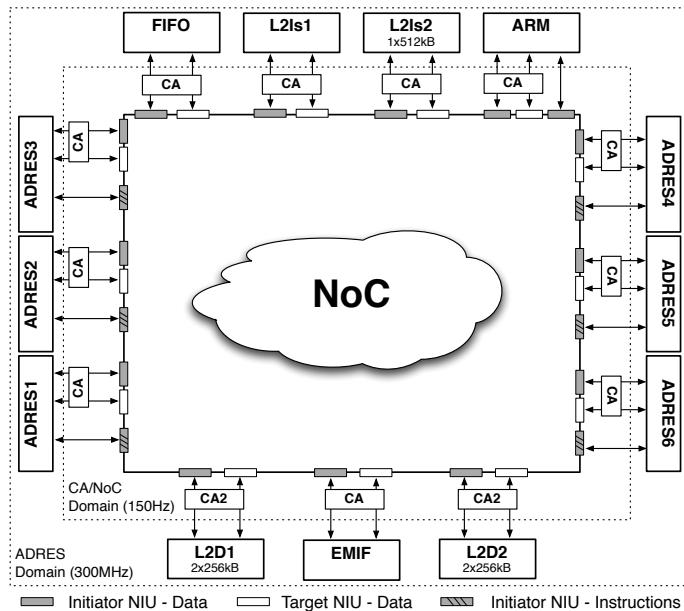


Figure 3.7 – Architecture of the 3MF MPSoC platform [6]

In summary, the problem we consider is to place 12 blocks while taking into account several (5) criteria. We will also consider a scenario where the blocks can be placed on 1 up to 5 tiers. The input data will be processed to generate partitions. Those output data will be encoded using the matrix model following the example shown in Table 3.2. They will be generated through a multi-objective optimization.

As explained earlier, we will use a metaheuristic to approximate the Pareto optimal frontier. For that purpose, we choose to use NSGA-II [5] as a proof of concept.

Table 3.10 – "Data split scenario" bandwidth requirements

Source	Target	Bandwidth (MB/s)
FIFO	EMIF	39.6
EMIF	ADRES <i>i</i>	6.6
L2D1	ADRES <i>i</i>	26.4
L2D2	L2D1	52.7
ADRES <i>i</i>	FIFO	1.2
ADRES <i>i</i>	L2D2	6.6
ADRES <i>j</i>	L2Is1	300
ADRES <i>k</i>	L2Is2	300

Index: $i, j, k \in \mathbb{N}^+$;
 $1 \leq i \leq 6; 1 \leq j \leq 3; 4 \leq k \leq 6$

Table 3.11 – Bandwidth input matrix

S	T	B	T	B	T	B	T	B	T	B	T	B	T	B
1	7	1.2	9	6.6	10	300	0	0	0	0	0	0	0	0
2	7	1.2	9	6.6	10	300	0	0	0	0	0	0	0	0
3	7	1.2	9	6.6	10	300	0	0	0	0	0	0	0	0
4	7	1.2	9	6.6	11	300	0	0	0	0	0	0	0	0
5	7	1.2	9	6.6	11	300	0	0	0	0	0	0	0	0
6	7	1.2	9	6.6	11	300	0	0	0	0	0	0	0	0
7	12	39.6	0	0	0	0	0	0	0	0	0	0	0	0
8	1	26.4	2	26.4	3	26.4	4	26.4	5	26.4	6	26.4		
9	8	52.7	0	0	0	0	0	0	0	0	0	0	0	
12	1	6.6	2	6.6	3	6.6	4	6.6	5	6.6	6	6.6		

S: source block ID
(T, B): target block ID and required bandwidth

Table 3.12 – Available technologies input matrix example

Technology (nm)				
90	60	45	32	22

The algorithm was run from a sample of 10 000 generated solutions from 1 up to 5 tiers. This size of random solutions is chosen arbitrarily since it is actually quite difficult to estimate the size of solution space, due to the number of degrees of freedom and the heterogeneous nature of the criteria. Also, taking too few solutions (e.g. 100) is not interesting since we have empirically observed that our algorithm will take a

longer time to begin to converge. 10 000 randomly-generated solutions seems to us a good compromise of time and workable solutions.

Results and their use for a designer

The simulations have been carried out on an Intel Core i5 2.30 GHz, 4 GB DDR3 SDRAM (runtime of 60 minutes, as explained in Section 3.4.2) and the presented results are the synthesis of 5 independent experiments. The optimization was done for three objectives (so that we can visualize the design space) and the main results are given in Figure 3.8 (interconnection length-cost projection) and Figure 3.9 (3D plot). Two conclusions can be drawn from that figure:

- The [10; 20] range values for the IL criteria: a small enhancement of the IL value leads to a large increase of the cost so the interest for a design with more than 4 tiers seems low.
- The [260; 280] range values for the cost criteria: a small increase of the price can give a large enhancement of the performance. A designer might consider accepting a slightly higher price for a sensitively better performance, knowing that this information can be quantified with an accurate model. Indeed, with the estimate model that we propose, a small 10% increase of the cost can decrease the IL by 60%.

These results did not take into account the degree of freedom of aspect ratio. If we go further by releasing a degree of freedom and allowing varying aspect ratios, we can have the Pareto front shown in Figure 3.10. This figure shows the Pareto front from Figure 3.8 (without aspect ratio, symbol: ·) alongside with a new Pareto front (with aspect ratio, symbol: +).

As expected, the Pareto front given when considering varying aspect ratios is globally better. Furthermore, by comparing the two graphs, we can see an interesting area where the two frontiers begin to merge at the cost value 350. This means that, in that area, it is not interesting to take the aspect ratio into account as the solutions will not necessarily be better. Once again, these kind of information can be important in the design of an IC and yet they would not be available with the current design flows since only a small number of possibilities are explored. Indeed, due to the sequential nature of the current design flows, such degrees of freedom are not even tried since they dramatically increase the duration of each optimization loop.

3.5.2 Case study 2: scalability

In the previous section, we have shown how a circuit can be modelled in 3D-SIC to apply a multi-objective optimization and run simulations based on the 3MF platform. This case study remains however limited as it contains only 12 blocks. In order to validate the results obtained previously, we will use a more realistic case

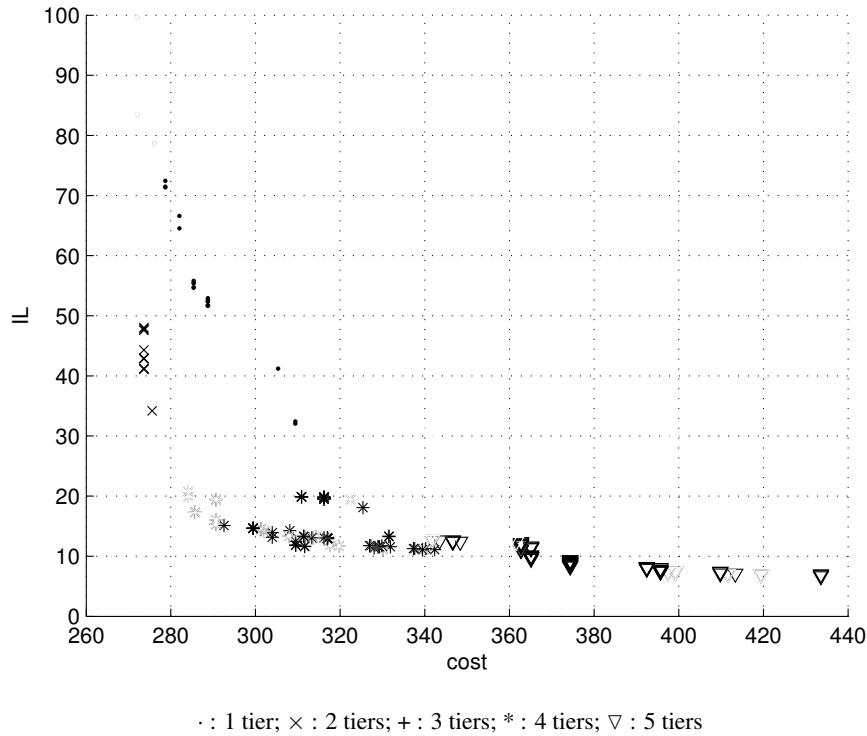


Figure 3.8 – IL-cost projection view of the Pareto frontier

study. It will be based on the 3MF platform where we will apply a scalability effect.

We will consider 24 ADRES processors instead of 6 and also separate the L1 cache memory from each ADRES core into L1 data cache memories and L1 instruction memories (total of 48 L1 memories blocks), assuming their size will be 10% of the related L2 memories. Since there is one L2D1 and one L2D2 data cache memory for 6 cores and one L2Is instruction memory for 3 cores in the original design, we will have 16 blocks of L2 memories. We will keep the FIFO and EMIF blocks from the original model. This will thus increase the total number of functional blocks to 90, which is a realistic number to work with. In addition, we will also consider size variability that will bring a form of functional heterogeneity to the circuit: $\pm 30\%$ for the ADRES cores and $+100\%$ for the L1 and L2 memories. This will give the input matrix given in Table 3.13.

For the communication requirements, the only change compared to the original case is that the exchanges between the ADRES processors and the L2 cache memories will now transit through the L1 cache memories. In order to simplify the neces-

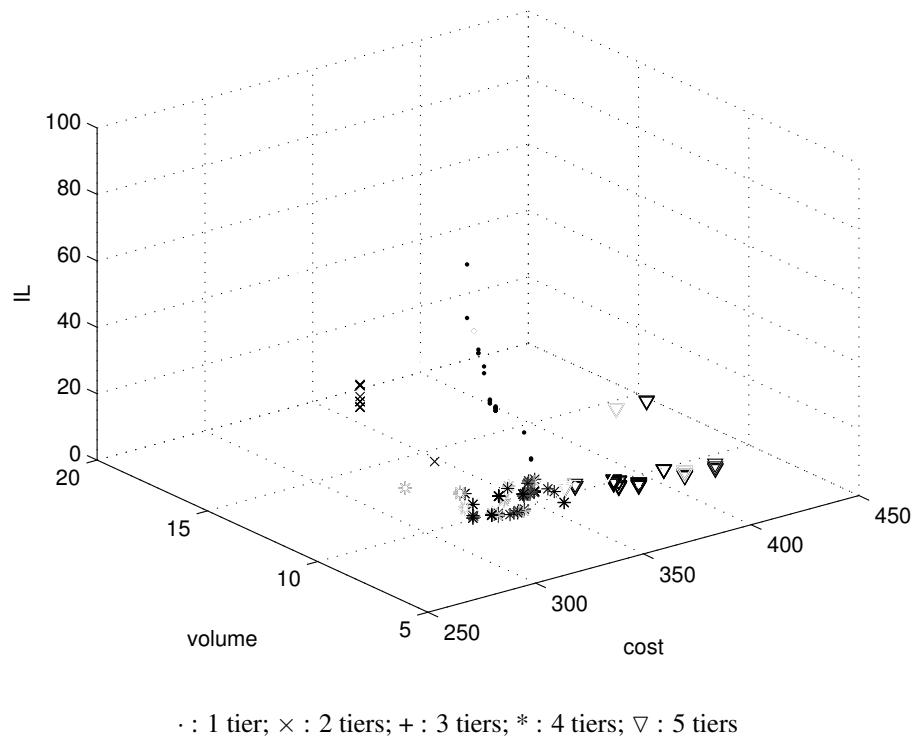


Figure 3.9 – 3D view (interconnection length-volume-cost) of the Pareto frontier

Table 3.13 – Scalability case input matrix

Component	ID	Size (90 nm)	Min aspect ratio	Size variability
ADRES core 1~24	1~24	17.27 mm ²	0.5	±30%
L1D	25~48	0.67 mm ²	0.5	+100%
L1Is	49~72	0.66 mm ²	0.5	+100%
L2D1#1~4	73~76	6.74 mm ²	0.5	+100%
L2D2#1~4	77~80	6.74 mm ²	0.5	+100%
L2Is1~8	81~88	6.62 mm ²	0.5	+100%
FIFO	89	0.54 mm ²	0.5	0
EMIF	90	0.66 mm ²	0.5	0

ID: Component identification number

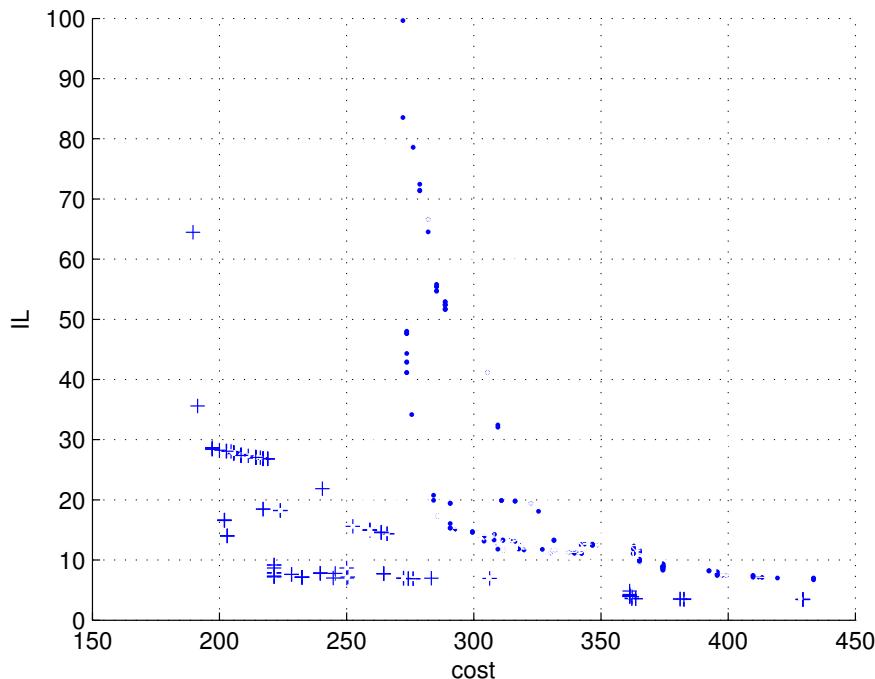


Figure 3.10 – IL-cost projection view of the Pareto frontier (with and without aspect ratio)

· : Pareto front without aspect ratio; + : Pareto front with aspect ratio

sary bandwidth between the ADRES cores and the L1 memories, we will assume a bandwidth 64 times bigger than for the L2. These data are summarized in Table 3.14 (the full matrix model can be found in Appendix II).

Simulation and validation procedure

In order to validate the methodology, we will run the simulations of this case study to produce partitions with 1 to 3 layers. These have been carried out on an Intel Core i5 2.30 GHz, 4 GB DDR3 SDRAM (runtime of 60 minutes) and the presented results are the synthesis of 5 independent experiments.

First, we will mimic the current design flow by performing a mono-objective optimization (minimizing the interconnection length). Then we will compare the obtained solution to other 2- and 3-tiers 3D-SIC that have been produced with a multi-objective approach.

Table 3.14 – Scalability case bandwidth requirements

Source	Target	Bandwidth (MB/s)
FIFO	EMIF	39.6
EMIF	ADRES core 1~24	6.6
ADRES core 1~24	L1D1~24	1690
ADRES core 1~24	L1Is1~24	19200
L2D1#1	L1D1~6	26.4
L2D1#2	L1D7~12	26.4
L2D1#3	L1D13~18	26.4
L2D1#4	L1D19~24	26.4
L2D2#1~4	L2D1#1~4	52.7
ADRES core 1~24	FIFO	1.2
L1D1~6	L2D2#1	6.6
L1D7~12	L2D2#2	6.6
L1D13~18	L2D2#3	6.6
L1D19~24	L2D2#4	6.6
L1Is1~3	L2Is1	300
L1Is4~6	L2Is2	300
L1Is7~9	L2Is3	300
L1Is10~12	L2Is4	300
L1Is13~15	L2Is5	300
L1Is16~18	L2Is6	300
L1Is19~21	L2Is7	300
L1Is22~24	L2Is8	300

After simulations, we obtain the following results for the mono-objective optimization of a 2D-IC and a 2-tiers circuit (we will denote them by A , B and C):

Alternative	Number of layers	Interconnection length
A	1	6.2737
B	2	5.2357
C	3	4.0438

Without surprise, we observe that the 3D-SICs perform better on that criterion than a 2D-IC, which validates the consistency of the model compare to what is expected in reality.

Let us compare A , to the 2- and 3-tiers circuits (respectively denoted D and E) obtained with a multi-objective optimization. We will make the comparison with circuits that are the closest to A with regards to the interconnection length criterion, in order to show the added value of working with a multi-criteria paradigm and simultaneously taking several criteria into account:

Alt.	Number of layers	Interconnection length	Cost	Volume	Clock position	Thermal dissipation
A	1	6.2737	$1.2178 \cdot 10^9$	$2.4356 \cdot 10^7$	$3.3584 \cdot 10^4$	81
D	2	6.2099	$1.2304 \cdot 10^9$	$2.4877 \cdot 10^7$	$3.8250 \cdot 10^4$	280
E	3	6.2205	$1.1495 \cdot 10^9$	$2.3589 \cdot 10^7$	$3.1095 \cdot 10^4$	459

As expected, except on the interconnection length criterion, the alternative D is dominated by A on all the criteria. Analysing the circuit E is more interesting though as we can observe that E dominates A on all criteria except for the thermal dissipation. This may seem surprising, especially for the cost and the volume criteria, however it can be explained by looking at the circuits themselves. In Figure 3.11, we can see the alternative A with a surface of $\sim 35 \times 35 \text{ mm}^2$ whereas for E, in Figure 3.12, the surface is $\sim 20 \times 20 \text{ mm}^2$ on 3 layers which is in total less than $35 \times 35 \text{ mm}^2$. This is not a big difference but it can show when it is interesting to design 3D circuits instead of 2D-ICs since the cost and the volume depend on the surface of a circuit. So this explains why E might perform better on these criteria. The clock source criterion is also better thanks to the smaller surface while the thermal dissipation is unsurprisingly worse due to the inherent greater power density of 3D-SICs.

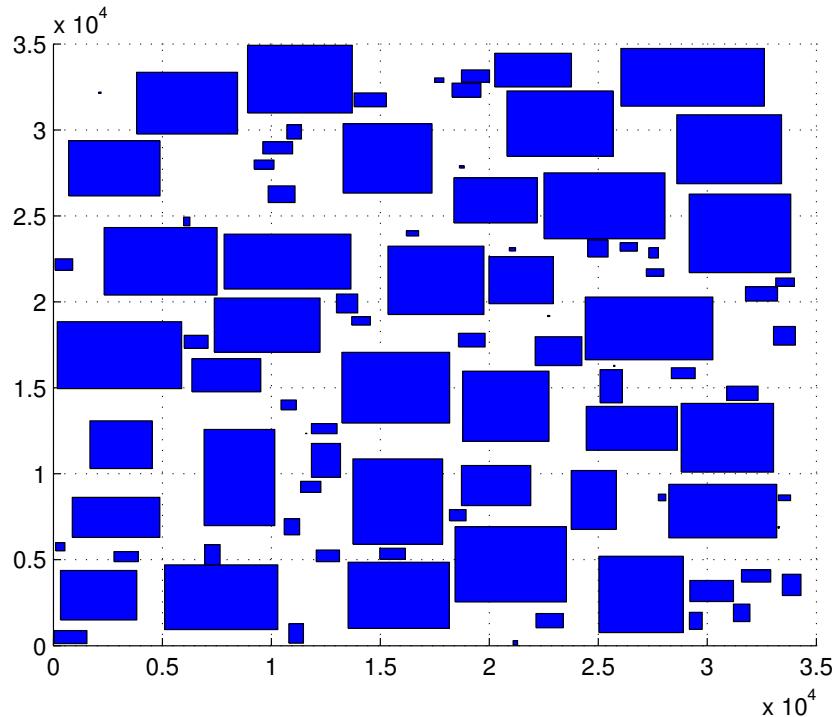


Figure 3.11 – Extended 3MF platform (1 layer)

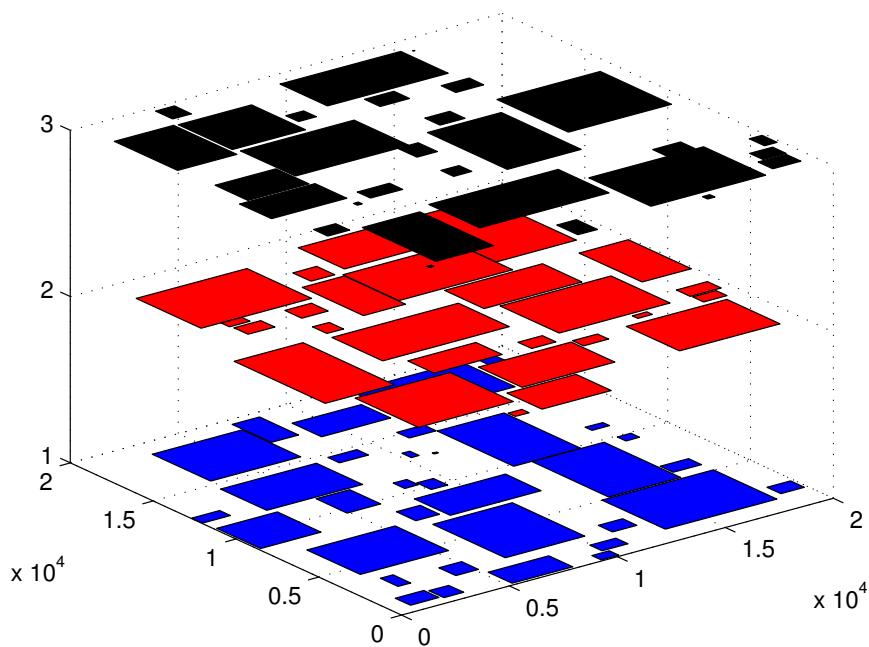


Figure 3.12 – Extended 3MF platform (3 layers)

Now let us compare B and C to D and E with a multi-criteria point of view:

Alt.	Number of layers	Interconnection length	Cost	Volume	Clock position	Thermal dissipation
B	2	5.2357	$1.2267 \cdot 10^9$	$2.4914 \cdot 10^7$	$4.5687 \cdot 10^4$	351
C	3	4.0438	$1.1495 \cdot 10^9$	$2.3681 \cdot 10^7$	$6.7234 \cdot 10^4$	593
D	2	6.2099	$1.2304 \cdot 10^9$	$2.4877 \cdot 10^7$	$3.8250 \cdot 10^4$	280
E	3	6.2205	$1.1495 \cdot 10^9$	$2.3589 \cdot 10^7$	$3.1095 \cdot 10^4$	459

B and C are the best uni-criterion 2- and 3-tiers circuits for the interconnection length while D and E are solutions close to A for that criterion. We can see that none of these alternatives dominate any other. Let us analyse these alternatives criterion per criterion. C is the best circuit for the interconnection length but if that criterion was not taken into consideration, C would be dominated since it is the only criterion where it performs well compared to the others. For the cost and the volume, C and E are the better than B and D for the reasons explained previously (smaller surface). As for the thermal dissipation the 2-tiers circuits B and D unsurprisingly perform better than the 3-tiers circuits.

If a mono-objective optimization was used, only the interconnection length would be considered and one would choose the circuit C as it holds the best score for this criterion while it performs rather badly on the clock position and thermal dissipation. E has better score in terms of volume, clock position and thermal dissipation but is not as good on the interconnection length criterion. If the thermal dissipation was the main criterion, the 2D circuit A would be the best choice though it can have a higher cost.

In a multi-criteria point of view, none of these circuits dominate any others so making a choice is not trivial. It would depend on the importance of each criterion compared to the others. We will show in Chapter 4 how these results can be exploited in order to help a designer choose in a transparent process. Let us finally note that these analyses have to be moderated within the limits of our model as, even if the criteria evaluation is consistent in an ordinal point of view, it is not physically accurate so quantitative comparisons would be imprecise compared to what would be expected in reality.

With these two case studies, we have shown that the proposed methodology can provide qualitative and quantitative information that would not be available with classical tools and that it can be applied for scaled problems that include degrees of freedom that are usually not considered, such as the form factor of the blocks and the heterogeneity. We have then analysed the solutions with a more global multi-criteria point of view and have shown that additional information is available to compare the solutions and making a choice among this small set of 5 alternatives is not a trivial task as it would seem to be when applying a uni-criterion paradigm.

In the next section we will analyse how robust the proposed methodology is by computing its performance with the indicators presented in Chapter 2 and in the next chapter, we will show how the obtained results can be exploited by using multi-criteria decision aid that would ease the choice of a designer.

3.6 Robustness of the methodology

In this section we will take a deeper look at the results of the multi-objective optimization steps. We will analyse the properties of the design space in order to have the view over the convergence and the robustness of our methodology the classical performance indicators used in the field (see Chapter 2).

The following results have been obtained with 5 independent runs. The set of non-dominated solutions over all the simulations will constitute the reference set R for the *epsilon* and the hypervolume indicators. Also, these results have been simulated with all the five criteria presented in Section 3.3 instead of only the three first for the case study of Section 3.5.1.

3.6.1 Contribution indicator

The Table 3.15 and the Figure 3.13 show the evolution of the averaged contribution indicator over the iterations for the 5 runs. We see that for the first iterations, $Cont(PO_i/PO_{i-1})$ is greater than 0.5, which means that the algorithm does indeed improve the solutions, then for the last iterations, the indicators are lower than 0.5 which means that there is a convergence.

Iteration	$Cont(PO_i/PO_{i-1})$
1	0.7626
2	0.8510
3	0.8917
4	0.8788
5	0.8295
...	...
38	0.4522
39	0.3870
40	0.2369

Table 3.15 – Evolution of the contribution indicator

3.6.2 Spread indicator

The Figure 3.14 shows the results of the spread indicator I_s function of the neighbourhood indicator σ (all the 5 runs share the same graph shape). We see that the Pareto front is well spread: if we consider $I_s \geq 0.9$ we have $\sigma < 0.35$ in average for the 5 runs (normalized values), so we can consider that the algorithm produces a well-spread approximation of the Pareto front.

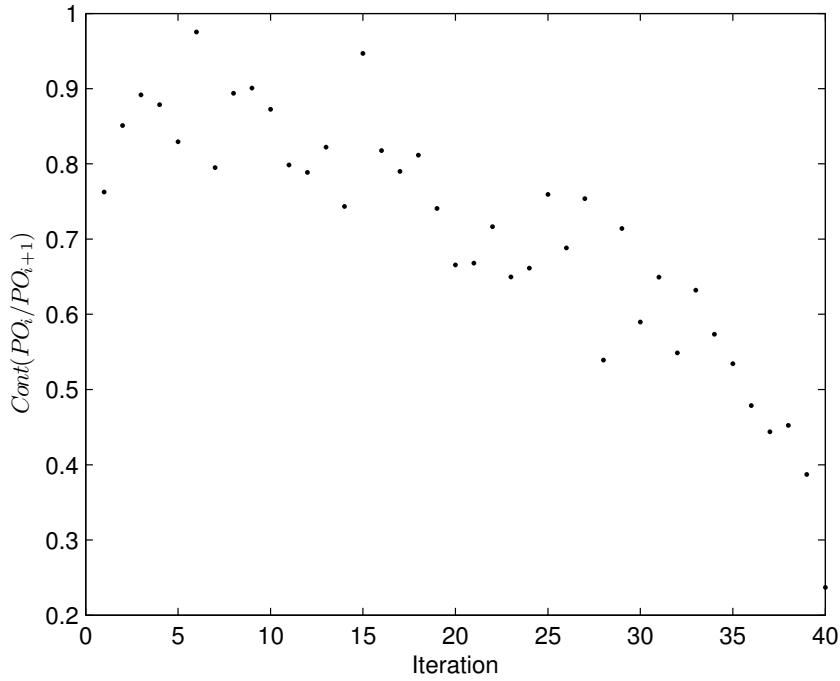


Figure 3.13 – Evolution of the contribution indicator

3.6.3 Binary ϵ -indicator

The non-dominated set computed from all the runs will constitute the reference set R that will serve to show the evolution of the ϵ -indicator over time. This evolution is shown in Table 3.16 (for averaged values after each 10 iterations). We can see that in the first iterations, $I_\epsilon(A, R) > 1$ and $I_\epsilon(R, A) \approx 1$ which means that the front is improved while in the last iterations, $I_\epsilon(A, R) > 1$ and $I_\epsilon(R, A) > 1$ which shows convergence.

A comparison of the binary ϵ -indicators between each experiment (after convergence) is also given, in Table 3.17. We can see that $I_\epsilon(A, B) > 1$ and $I_\epsilon(B, A) > 1$ which indicates that neither A weakly dominates B nor B weakly dominates S . This means that the generated front is consistent from one experiment to another.

Also, in Table 3.18 are given the ϵ -indicator between iterations of an experiment (the same observations apply for the other runs). We can see that in the first iterations, the front is always improved ($I_\epsilon(A_i, A_{i-1}) > 1$ and $I_\epsilon(A_{i-1}, A_i) \leq 1$) while in the last iterations, it begins to converge ($I_\epsilon(A_i, A_{i-1}) > 1$ and $I_\epsilon(A_{i-1}, A_i) > 1$).

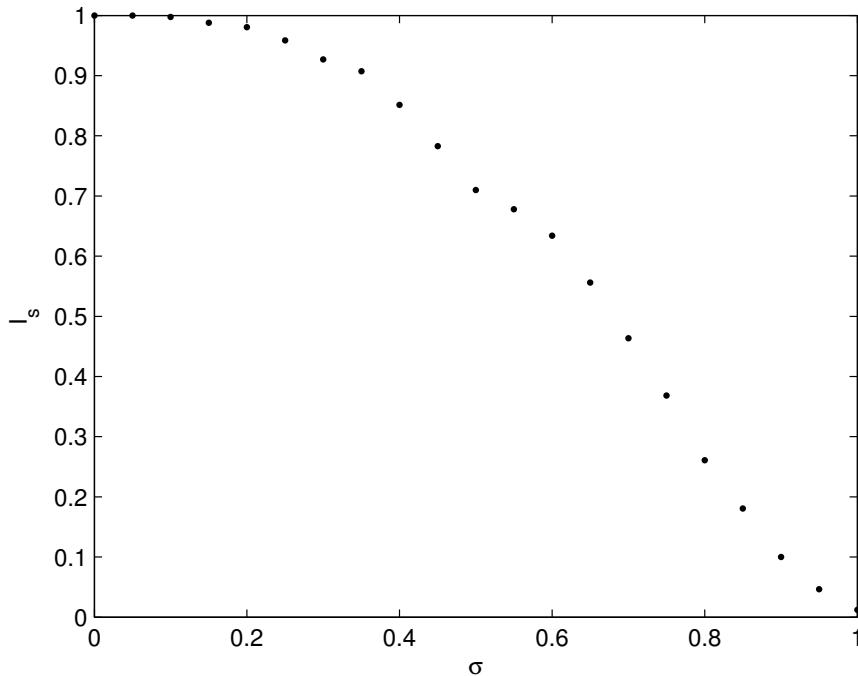


Figure 3.14 – Spread indicator I_s function of the neighbourhood parameter σ

Iteration	Averaged $I_\epsilon(A, R)$	Averaged $I_\epsilon(R, A)$
1	5.5255	1.0178
10	4.4307	1.0235
20	3.8102	1.1023
30	2.3614	1.1234
40	1.6569	1.2381

Table 3.16 – Evolution of the binary ϵ -indicator (averaged values compared to the reference set R) over time

3.6.4 Unary hypervolume indicator

The evolution of the hypervolume (averaged values) is given in Table 3.19 and in Figure 3.15. We can see that the value is (linearly) increasing over time, reaching convergence in the last iterations where the value stabilizes.

The result for each experiment is also given, in Table 3.20 and the used reference point is the worst point computed from all the sets for normalized data. As we can see, the values are rather consistent from one run to another.

$I_\epsilon(A, B)$	Run 1	Run 2	Run 3	Run 4	Run 5
Run 1	1	1.7270	1.3594	1.8664	1.2542
Run 2	1.5713	1	1.4122	1.7791	1.3420
Run 3	1.4737	1.8638	1	1.9268	1.3069
Run 4	1.3436	1.4564	1.2843	1	1.2365
Run 5	1.4214	1.7650	1.3918	1.7545	1

Table 3.17 – Comparison of the binary ϵ -indicators for each experiment

Iteration	$I_\epsilon(A_i, A_{i-1})$	$I_\epsilon(A_{i-1}, A_i)$
1	1.6674	1.1053
2	1.7223	1
3	2.4439	1
4	1.7477	1
5	2.0577	1
...
38	1.8788	1.4916
39	1.5344	1.8065
40	1.9862	1.6609

Table 3.18 – Comparison of the binary ϵ -indicators between iterations of the same experiment

Iteration	Averaged hypervolume
1	0.0574
10	0.0701
20	0.0876
30	0.0931
40	0.1036

Table 3.19 – Evolution of the unary hypervolume indicator (averaged values compared to the reference set R) over time

3.6.5 Density of the Pareto front - gaps in the frontier

Another indicator of the Pareto front structure is its density. Here we will measure the density by finding gaps in the frontier. This will be done by counting the number of solutions in the neighbourhood of another solution. Since the extreme distance between two solutions is 450.364 in average for the 5 runs (non-normalized),

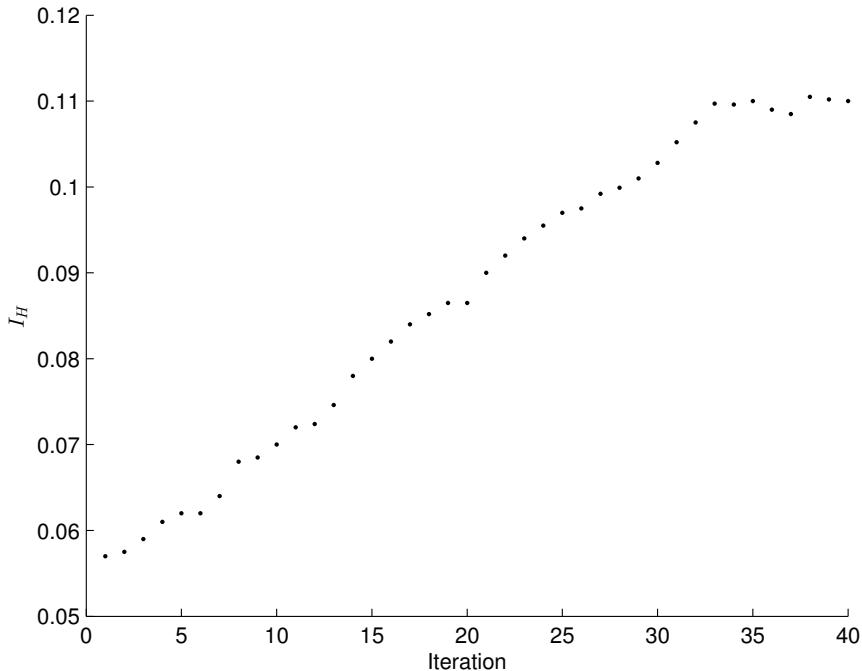


Figure 3.15 – Evolution of the unary hypervolume indicator (averaged values compared to the reference set R over time

Run	I_H
1	0.1171
2	0.1105
3	0.1015
4	0.1185
5	0.0939

Table 3.20 – Hypervolume for each experiment

we consider that an acceptable neighbourhood is twice the distance between two solutions if all the solutions were equidistant. We have thus a neighbourhood of about 2. This test has shown that there was always at least one solution near another one, even for a neighbourhood of 1, meaning that the algorithm can produce a sufficiently dense frontier.

3.6.6 Convexity indicator

The convexity is also an indicator of the structure, so analyses have been performed to determine the shape of the Pareto front. Globally, the Pareto front is not convex, as one may expect given the heterogeneous nature of the criteria and correlation between them. Also, since an approximate algorithm has been used, small gaps between solutions may exist that will not give a convex shape to the frontier.

3.6.7 A word on the scalability of the methodology

The previous results have been obtained with the case study shown in Section 3.5.1. Similar results can be obtained for the scaled case study of Section 3.5.2 which contains about a hundred blocks, and it has also been successfully tested for a few hundreds of randomly-generated blocks. Reaching several hundred to a thousand of blocks, the algorithm starts to take tens of minutes per iteration which shows its limits in terms of scalability even though convergence can still be shown.

3.7 Conclusion

In this chapter, we have proposed a 3D-SIC model that can take into account 5 different criteria to apply MOO with a more global multi-criteria analysis. To the best of our knowledge, current tools deal with a limited set of criteria (usually 3) and only perform trade-off analyses. We have performed simulation on two different case studies and the results have shown interesting information that can be relevant for a designer. First, using a multi-objective optimization methodology does not only consider all the criteria at the same time but also proceed to an extensive design space exploration which is rarely done with current tools. Second, the qualitative results shown here can give relevant information to the designer and they can even be quantified with a more accurate model. Third, the flexibility of MOO allows to easily consider new degrees of freedom, such as the form factor of the blocks and the heterogeneity, without having to change the paradigm. Finally, we have validated the methodology with a scaled case study and shown that a multi-objective optimization gives added values compared to a uni-criterion paradigm.

We have then shown that the proposed methodology has proved to be robust even if the problem contains criteria of heterogeneous nature. With the several indicators that we have analysed, we can conclude that the algorithm we used can show good properties of convergence, spread and density.

In the next chapter, we will show how these results can be exploited, with a multi-criteria point of view, to help a designer make a choice among a set of efficient solutions.

4

Results exploitation

Chapter abstract

In this chapter, we show how the obtained results with multi-objective optimization can be used to help a designer, using multi-criteria decision aid. We present how the solutions can be analysed, with a multi-criteria point of view. Then we discuss on their possible actual use and how difficult it is to make the industry adopt it. We then propose some hints to make it integrated into design flows as we do believe that a multi-criteria paradigm can help in design integrated circuits and that it can progressively be of use since designers will have to face greater and greater challenges.

Associated publications:

- N.A.V. Doan, D. Milojevic, F. Robert, Y. De Smet, "Using the PROMETHEE methodology for the design of 3D-stacked integrated circuits", *1st International MCDA Workshop on PROMETHEE (IMW2014)*, Bruxelles (Belgique), January 2014
- K. Lidouh, N.A.V. Doan, Y. De Smet, "PROMETHEE-compatible presentations of multicriteria evaluation tables", *International Journal of Multicriteria Decision Making*, to be published (minor revision)
- K. Lidouh, N.A.V. Doan, Y. De Smet, "PROMETHEE-compatible presentations of multicriteria evaluation tables", *2nd International MCDA Workshop on PROMETHEE (IMW2015)*, Bruxelles (Belgique), January 2015, to be published (accepted)

4.1 Introduction

In Chapter 3 we have defined the problem we are dealing with in this work, show how a 3D-SIC can be modelled, as well as simulation results based on multi-objective optimization. In Section 3.6, we have shown that the methodology can show good convergence and diversity properties even if the problem contains criteria of heterogeneous nature. In this chapter, we will discuss about how a designer could use these results and take advantage of a multi-criteria oriented methodology in the process of producing a 3D-SIC to, for instance, make a choice among the solutions of the Pareto frontier.

As explained in Chapter 2, once a Pareto front has been determined or approximated, the next step is to choose among this set of solutions. The simulations performed in the experiments of Section 3.6 have produced 804 individuals in the Pareto front and with such a large number of alternatives, choosing is not always trivial.

4.2 Preference modelling

One way to help decision makers to make their choice is to model their preferences, for instance with an outranking method. In the scope of this work, we will present the use of the PROMETHEE methodology as it has been developed in our department with an efficient software called D-Sight and has also shown good results in different fields [68]. In addition, using this method can be justified as the evaluations are quantified. Also, even if the number of alternatives is large, PROMETHEE does not require a large number of inputs from the decision maker, as opposed to some other methods such as AHP (see Chapter 2) where pairwise comparisons of alternatives can be asked to a DM.

4.2.1 Using the PROMETHEE methods

Building a PROMETHEE model

In order to use the PROMETHEE method, the decision maker has to inform about his preferences on the criteria, these being preference functions, indifference and preference thresholds and weights on the criteria (see Chapter 2). To illustrate this, we will use a PROMETHEE software called D-Sight that has been developed by Quantin Hayez and use the results of the simulations presented in Section 3.6. In those results, there are 804 alternatives in the Pareto front, evaluated on 5 criteria. The evaluation table can be found in Appendix I. Let us take a simple example of preference modelling in order to illustrate what kind of aid a multi-criteria analysis can provide.

First, we have to define preference functions for each criterion. For the interconnection distance, cost, volume and clock position, we will choose a V-shape function. For these criteria, using this function, the preference index will increase linearly until a preference threshold is reached. For the power dissipation, we will choose a U-shape function where there is no preference until an indifference threshold is reached. Indeed there can be no real problem if the difference between two circuits in terms of heat dissipation is low, and it can be directly problematic when this difference is high.

The thresholds will be set at a difference of 10% in the evaluations. For the weights, we will consider that the interconnection distance, the cost and the power dissipation are more important than the volume and the clock position, with the volume less important than the clock position. We will choose the weights by considering that the three first cited criteria share the same importance, 25% each, and the two remaining ones taking the last 25% (14% and 11% respectively). The robustness of this set of weights can be studied with a sensitivity analysis that will allow to confirm whether or not the variability of a weight will change the ranking. Let us note that it is possible to elicit the weights by answering simple questions, for instance with AHP.

A summary of all these data is given in Table 4.1. Of course, a more accurate model could be defined, however let us remind that the purpose here is to show how a multi-criteria analysis can give added information to designers.

Table 4.1 – PROMETHEE model

Criterion	Preference function	Indifference threshold	Preference threshold	Weight
Interconnection distance	V-shape	x	10%	25%
Cost	V-shape	x	10%	25%
Volume	V-shape	x	10%	11%
Clock position	V-shape	x	10%	14%
Power dissipation	U-shape	10%	x	25%

Now that a model has been proposed, let us analyse the results produced by D-Sight.

Multi-criteria analysis

PROMETHEE rankings D-Sight will do all the computations of the flows and PROMETHEE (I and II) rankings can be obtained, based on the preferences. A decision maker can make a choice based on these rankings, for example by choosing

the solution ranked first. In addition, other tools are available, that allow to have a transparent decision process and analyse the set of solutions to know why a given ranking is obtained. One of the most useful one is the GAIA plane which is illustrated in Figure 4.1 (for the sake of readability, the alternative names have been removed).

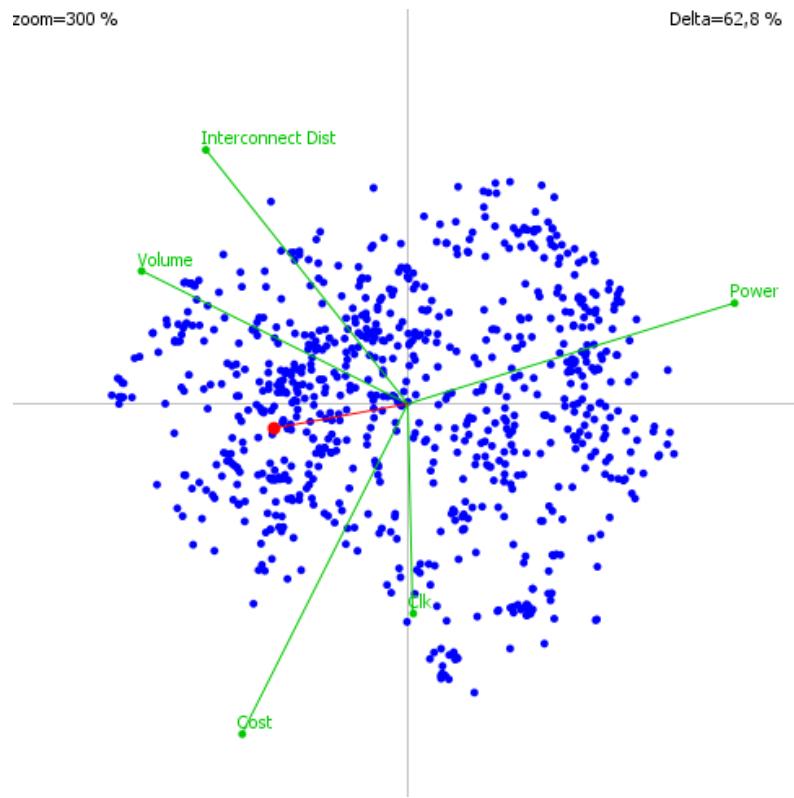


Figure 4.1 – GAIA plane of the case study

As a reminder from Chapter 2, the GAIA plane is based on the principal component analysis of the unicriterion net flows of the solutions and minimises the projection error of each alternative on it. Four distinctive visual information are shown:

1. The green axes that represent the projections of each criterion's axis.
2. The blue dots that represent the projection of each solution's uni-criterion net flow. The value of the uni-criterion net flow is read by projecting the point on the related criterion axis.
3. The red axis that represents the *decision axis* which is the projection of the set of weights and gives the decision direction.

4. The *delta* value that represents the percentage of kept information since there are projection errors.

The first observation that can be drawn is that the blue dots are at the same time well-spread and dense, which illustrate the conclusions of Section 3.6. This also means that each criterion is well-represented in terms of solutions.

Second, let us take a look at the information that is provided by the criteria axes (green). This is shown in Figure 4.2.

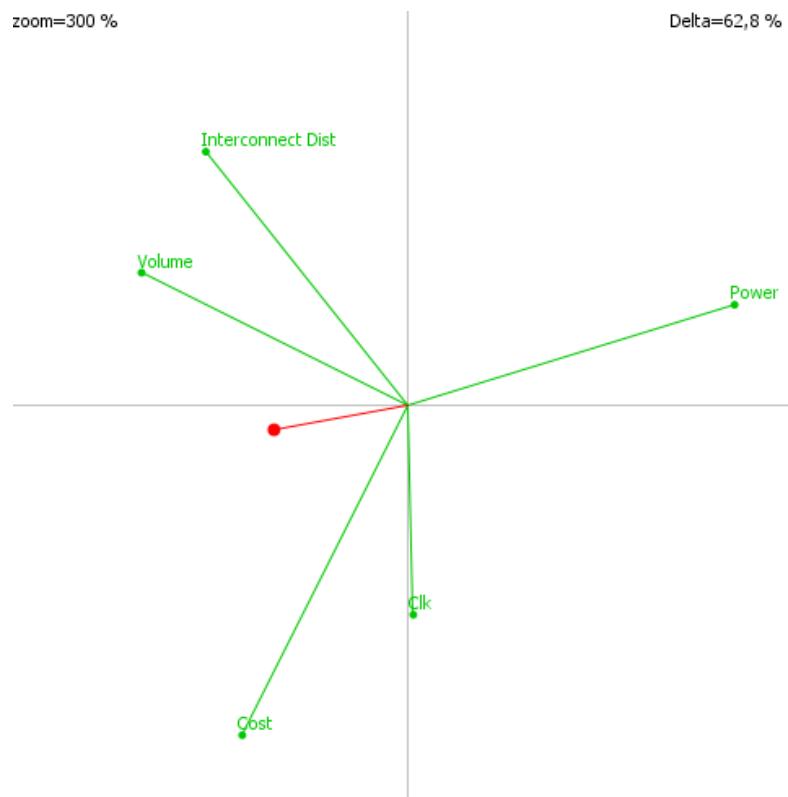


Figure 4.2 – GAIA plane of the case study (criteria axis only)

From the GAIA plane, we can observe how the criteria are related between each other. Indeed, criteria axes that have opposite directions are conflicting, whereas criteria with the same direction are in synergy. In the present case, we can see that the criteria of interconnection distance, power dissipation and cost are conflicting, which reflects the design reality. Also, the volume criterion shares the same direction as the interconnection distance criterion which is normal as reducing the interconnection distance will also tend to reduce the circuit volume. These observations also confirm that the defined model is indeed consistent with the reality.

Finally, let us have a view at the information provided by the decision axis. As explained previously, the decision axis represent the criteria weights and therefore gives the decision direction. Indeed, the alternatives with the highest net flow score will have their furthest projection on that axis, in the direction of that axis (see Figure 4.3). This visually represents the PROMETHEE II ranking, provided that the *delta* value shows that enough information has been kept with the projection. In this case, this value is relatively low (62.8%) which means that a lot of information has been lost with the projection, that can lead, for instance, to PROMETHEE II ranking errors with the visual projections compared to the ranking obtained with the computed net flows.

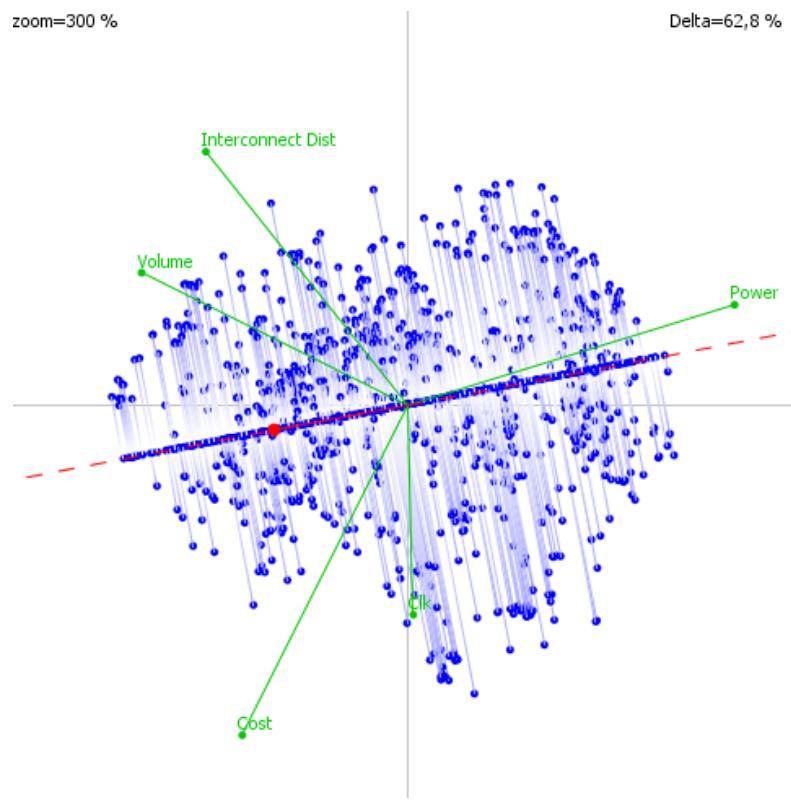


Figure 4.3 – GAIA plane of the case study (with decision axis projection)

Robustness analysis with stability intervals Another tool that can help decision makers is the robustness analysis that will allow them to know how stable a solution is, given the provided preference model. It is based on stability intervals on the weights where the first-ranked alternative will not change. This tool can be useful

as there can be uncertainties on the values given for the weights. For the considered model, the stability intervals are shown in Table 4.2. We can observe that the first-ranked solution is relatively robust with all the criteria weights spanning on rather large intervals. This means that small uncertainties will not affect the ranking of the first alternative.

Table 4.2 – Stability intervals (level 1)

Criterion	Min weight	Value	Max weight
Interconnection distance	5.73%	25%	50.00%
Cost	3.37%	25%	36.38%
Volume	0.00%	11%	23.85%
Clock position	2.06%	14%	43.06%
Power dissipation	17.85%	25%	68.21%

As we can observe, the PROMETHEE methodology can help a decision maker facing choices and provide a transparent process. While the tools have been developed in order to be simple to use and analyse, the main difficulty is to model the preferences accordingly with a designer's needs. As the specifications required for a design cannot translate easily into preference information, establishing a preference model is not a trivial task and this will therefore need further investigations to adapt this methodology for designers.

4.3 Constraint modelling

To ease decision making, it is also possible to model constraints in order to eliminate unrequired alternatives and reduce the number of solutions, which will ease the choice process. For that purpose, we have developed a visual interface where a decision maker can introduce constraints to be fulfilled and see directly the remaining solutions that fit these requirements. The general interface is shown in Figure 4.4.

The sliders are used to define the constraints. Two graphs are represented to visualise the solutions. The first one uses parallel axes. Let us note that, in order to use this representation effectively, we have normalised the evaluations between 0 and 1. The second graph is a PCA plane where the evaluations are projected (not the unicriterion net flows). It can help a decision maker to more easily see in which direction the solutions are.

In Figure 4.5, an example is shown where alternatives have been filtered. For illustration purposes, we have arbitrarily chosen values by considering normalized constraints of 0.5 for the interconnection length, 0.3 for the cost, 0.8 for the volume,

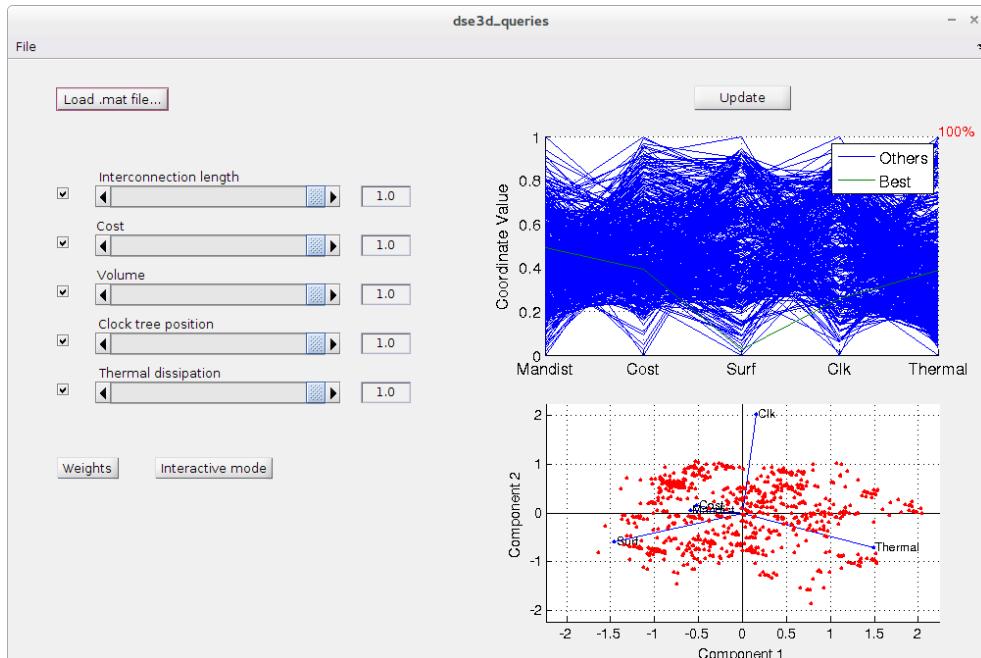


Figure 4.4 – Constraints modelling (without filtered alternatives)

0.7 for the clock source position and 0.4 for the thermal dissipation (taking lower values means adding more constraints). Alternatives that do not reach these constraints are removed from the visualisation and we can see that from 804 potential solutions, there are 5 remaining possibilities which can ease the choice process.

This software also allows the see the profile of the best profile given a set of weight for the criteria, by using a weighted sum. These weights can also be elicited with the AHP that has been implemented within the interface.

This tool has been developed under Matlab and can be used for any problems that need to filter unrequired solutions with constraints to ease a decision process.

The users will eventually have to be aware that the evaluations have to be normalized between 0 and 1 in order to have an effective representation with the parallel axes. So, modelling the constraints can be more difficult than with the real evaluations. In the same way as preference modelling, the specifications cannot be easily translated into constraints so this would also require further investigations to adapt this method for designers.

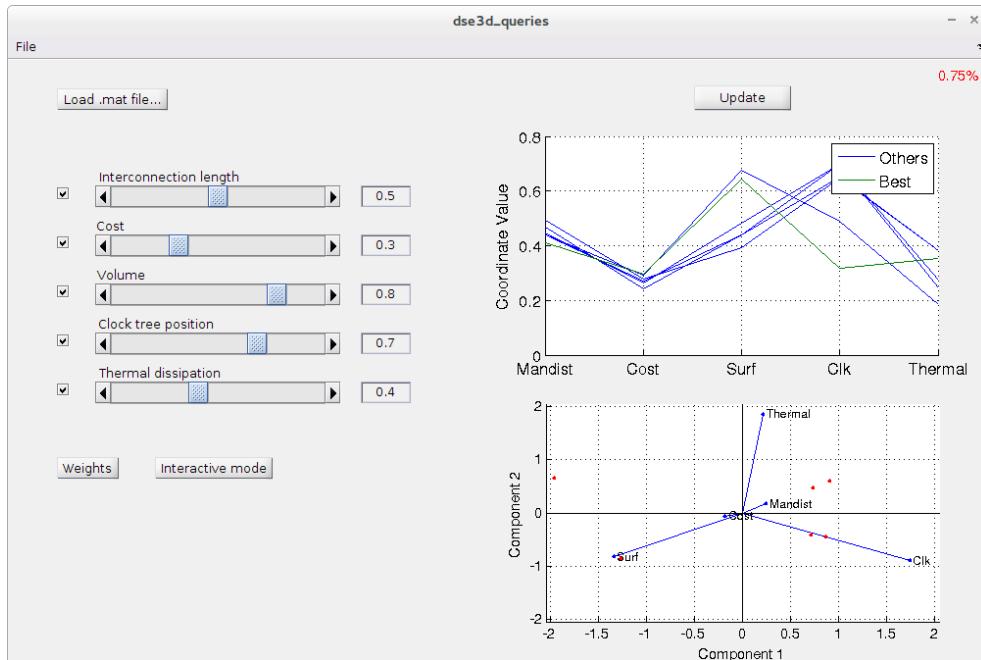


Figure 4.5 – Constraints modelling (with filtered alternatives)

4.4 Pertinently representing multi-criteria information in evaluation tables

Another way to help decision making could be to enrich evaluation tables with multi-criteria information. We have proposed a contribution with that purpose in [76]. However, as it is difficult to find a direct application of this methodology in microelectronic design, this work will only be cited and reproduced in Appendix III.

4.5 On the use of a multi-criteria paradigm in microelectronic design

As described in Chapter 1, the multi-criteria paradigm is rarely used in the field of IC design; at best, trade-off analyses are performed. To our knowledge, more global multi-criteria analyses have not been carried out yet.

When discussing with design experts, it is quite interesting to see how they easily understand the stake of the MCDA paradigm and how it would be able to help designers facing IC development challenges. However, it is more difficult to make

them adopt this approach for three main reasons that have appeared throughout several discussions:

1. "It is not how we optimize circuits" seems to be one of the most frequent statements. Indeed, the industry follows a uni-criterion paradigm and is not used to first explore several possible solutions and then determine good compromise solutions. The designers will generally decide about an architecture and try to optimize it (following a uni-criterion paradigm) to fulfil the specifications.
2. Designers can understand how preference modelling work, however they are not used to answer questions about indifference/preference thresholds or criteria weights as they receive specifications to achieve. This would need a change in how the design of a circuit is approached and how specifications are formulated.
3. The design space exploration is based on performance estimation. While our model can provide consistent ordered information, the evaluations are not accurate. Therefore, modelling preference can be a more difficult task since it requires the designers to work with (assessed) values that are different from real specifications.

As we can observe, the main reason of difficulties to adopt a multi-criteria paradigm lies in the lack of knowledge about this approach. This will need a deep work in all the steps in a design flow, from how the specifications are defined to the optimization processes. Specifications are currently more and more difficult to fulfil as the industry is nearing the limits of the present technologies. Changing how they are formulated, with therefore adequate methodologies, might help overcome this problem.

Also, (uni-criterion) optimization processes are nowadays more and more time-consuming (weeks to months). This can be seriously problematic in economical terms as a circuit will require more man-years. With this work, we have shown that applying a multi-objective optimization for design space exploration can shorten the design time. These simulations only last hours to days and can already give to designer assessments about the optimization of a circuit, which might lead to shorter optimization processes.

Finally, with the results we have obtained, we do believe that the multi-criteria analysis can aid designers when facing design challenges and allow them to make more transparent choices.

4.6 Conclusion

In this chapter, we have presented how the results of a multi-criteria approach can be exploited for designers. We have shown two ways to help in a decision process

and cited a third work. Then we have discussed about the adoption of this paradigm in the field of microelectronic design where we have proposed some possible hints. Although the results we have obtained can provide relevant information to a designer that would not be available with current tools, their exploitation seems to be more complicated as it would require a change in how the industry works. Nevertheless, we do believe that a multi-criteria paradigm can help in design integrated circuits and that it can progressively be integrated into design flows since designers will have to face greater and greater challenges.

5

Conclusions

In this thesis, we have studied the applicability of multi-objective optimization and multi-criteria decision aid in the context of 3D-stacked integrated circuits design. In the past decades, the electronic industry has been following the Moore's law to improve the performance of integrated circuits. However, due to physical limitations appearing with the miniaturization of the transistors below a certain threshold, it will probably be impossible to follow this law in the future.

In order to overcome this problem, new technologies have emerged, and among them the 3D-Stacked Integrated Circuits have been proposed by the academic and industrial communities. 3D-SICs can bring numerous advantages in the design of future ICs but at the cost of additional design complexity due to their highly combinatorial nature, and requiring the optimization of several conflicting criteria. On the other hand, the multi-criteria approach has been discussed in the literature as a paradigm to adopt for solving numerous similar problems in different industrial fields.

We have therefore proposed to apply multi-objective optimization and multi-criteria decision aid for the design of 3D-SIC. First, we have defined the problem we tackle, the 3D partitioning with floorplanning estimation, with the considered criteria. We have proposed a model for 3D-SIC that include 5 criteria and degrees of freedom that are not usually considered such as the form factor of the blocks and the heterogeneity while to the best of our knowledge current tools use a limited set of criteria (usually 3) and only make trade-off analyses.

Simulations have been performed on a case study based on the 3MF MPSoC platform and the obtained results have shown that qualitative and quantitative infor-

mation, that would not be available with current tools, can be provided to designers. We have also validated the method by using a scaled case study with functional heterogeneity and have shown that a multi-criteria paradigm has added value compared to trade-off analyses that are performed with current tools. Indeed, we have shown that with a more global multi-criteria point view additional information is available to compare the solutions and making a choice among these alternatives is not a trivial task as it would seem to be when applying a uni-criterion paradigm.

We have then proved that the methodology and associated algorithms are robust even though the problem of designing 3D-SIC is complex, with criteria of heterogeneous nature. By using classical indicators of the field, we have demonstrated good convergence and diversity properties. Finally, we have shown how the obtained results can be exploited and how multi-criteria decision aid tools could help a designer by providing additional information.

Since the focus of this thesis is to show the applicability of a multi-criteria paradigm to the design of 3D-SIC, there remain several open perspectives. In particular, the accuracy of the model we proposed is yet to be improved. We have shown that MOO can provide quantitative information to designers. Currently, the values have been modelled to respect the order of the solutions in reality. Having a more precise model would allow to propose more realistic quantitative analyses.

In Chapter 4 we have presented some methods to exploit the results with multi-criteria decision aid methods. However, as discussed, applying MCDA for the design of ICs is not trivial as the industry is not used to this paradigm. This can be improved by studying how MCDA can be adapted to the microelectronic field in order to address its specific needs. For instance, it can be interesting to see how industrial specifications could be translated into a multi-criteria models.

Another line of research is the improvement of thermal dissipation criterion. In our model, we have only considered the peak output power as computing a thermal map can be time consuming. An improvement could be to develop a method that can quickly compute the thermal dissipation while limiting precision loss. Since this is one of the most critical issues of 3D-SICs, this topic is actually a research field on its own with several developments of thermal-aware partitioner/floorplanner. One can cite the works in [23, 24, 71].

In our algorithm, we produce initial solutions randomly. This is one of the two possibilities to generate solutions. A further development is to adopt a constructivist approach for building partitions, for instance with a GRASP-type algorithm [74, 77]. This would require to develop a multi-objective version of GRASP adapted for the design of 3D-SICs. Such an algorithm would allow to dispose of "good" solutions before beginning the design space exploration and may reduce simulation times.

Within the scope of this thesis, we have focused our developments at the logical level of a design flow. With the obtained results, we believe that a multi-criteria methodology can be transposed at other levels such as the architecture level or even

the physical level since a uni-criterion paradigm is used at each step. This should give added information to designers and help them facing the growing complexity of ICs.

A particularly important topic that we have not considered in this work is the applications that will run on a platform. This is actually related to hardware/software (HW/SW) co-design which is a research topic on its own. HW/SW co-design aims to match the right software on the right hardware platform in order to take the best out of a design [78]. Considering that HW/SW co-design can be summarized to a multi-criteria combinatorial problem, applying MCDA to that field would already be an improvement. Furthermore, integrating co-design to a performance assessment model will improve it since the estimations will be more precise. For instance, the consumption of a circuit (and its thermal dissipation) will be more accurate as it depends on the application that is run on a platform.

Despite the many questions we open, we still provide an answer to the main research question we posed in the introduction: "Is it possible to apply multi-objective optimization and multi-criteria decision aid to the design of 3D-stacked integrated circuits?". As we have shown, the obtained results provide a first positive answer to this question.

Bibliography

- [1] R. X. Cringely. (2013) Breaking moore's law @ONLINE. [Online]. Available: <http://betanews.com/2013/10/15/breaking-moores-law/>
- [2] R. Kirchain and L. Kimerling, "A roadmap for nanophotonics," *Nature Photonics*, vol. 1, pp. 303 – 305, jun 2007.
- [3] D. Noice and V. Gerousis, "Physical Design Implementation for 3D IC - Methodology and Tools." Cadence Keynote, 2010. [Online]. Available: http://ispd.cc/slides/slides10/4_02.pdf
- [4] S. Al-Sarawi, D. Abbott, and P. Franzon, "A review of 3-d packaging technology," *Components, Packaging, and Manufacturing Technology, Part B: Advanced Packaging, IEEE Transactions on*, vol. 21, no. 1, pp. 2 –14, feb 1998.
- [5] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multi-objective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 182–197, 2000.
- [6] D. Milojevic, L. Montperrus, and D. Verkest, "Power dissipation of the network-on-chip in multi-processor system-on-chip dedicated for video coding applications," *Journal of Signal Processing Systems*, p. 15, June 2008.
- [7] J. Brans and B. Mareschal, *PROMETHEE-GAIA. Une Méthodologie d'Aide à la Décision en Présence de Critères Multiples*. Paris, France: Ellipses, 2002.
- [8] S. Borkar, "Design perspectives on 22nm cmos and beyond," in *Design Automation Conference, 2009. DAC '09. 46th ACM/IEEE*, July 2009, pp. 93–94.
- [9] V. Zhirnov, I. Cavin, R.K., J. Hutchby, and G. Bourianoff, "Limits to binary logic switch scaling - a gedanken model," *Proceedings of the IEEE*, vol. 91, no. 11, pp. 1934 – 1939, nov 2003.
- [10] D. Milojevic, R. Varadarajan, D. Seynhaeve, and P. Marchal, *PathFinding and TechTuning in Three Dimensional System Integration*, A. Papanikolaou, D. Soudris, and R. Radojcic, Eds. Springer, Nov 2011. [Online]. Available: <http://www.springer.com/architecture+%26+design/architecture/book/978-1-4419-0961-9>

- [11] S. Tans, A. Verschueren, and C. Dekker, “Room-temperature transistor based on a single carbon nanotube,” *Nature*, vol. 393, no. 6680, pp. 49–52, 1998.
- [12] Y. Cui, Z. Zhong, D. Wang, W. U. Wang, and C. M. Lieber, “High performance silicon nanowire field effect transistors,” *Nano Letters*, vol. 3, no. 2, pp. 149–152, 2003. [Online]. Available: <http://pubs.acs.org/doi/abs/10.1021/nl0258751>
- [13] D. Goldhaber-Gordon, H. Shtrikman, D. Mahalu, D. Abusch-Magder, U. Meirav, and M. A. Kastner, “Kondo effect in a single-electron transistor,” *Nature*, vol. 391, no. 6663, pp. 156–159, Jan. 1998.
- [14] R. Patti, “Three-dimensional integrated circuits and the future of system-on-chip designs,” *Proceedings of the IEEE*, vol. 94, no. 6, pp. 1214–1224, June 2006.
- [15] S. Das, A. Fan, K.-N. Chen, *et al.*, “Technology, performance, and computer-aided design of three-dimensional integrated circuits,” pp. 108–115, 2004.
- [16] E. Beyne and B. Swinnen, “3D System Integration Technologies,” *Integrated Circuit Design and Technology, 2007. ICICDT '07. IEEE International Conference on*, pp. 1–3, May 2007.
- [17] A. V. Biest, D. Milojevic, and F. Robert, “Key enablers for next generation system-level design in microelectronics,” in *Proc. 10th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI), Orlando (Florida), 16-19 July 2006*, 2006.
- [18] E.-G. Talbi, *Metaheuristics : from design to implementation.* John Wiley & Sons, 2009.
- [19] E. Sicard and S. Dhia, “An illustration of 90nm CMOS layout on pc,” in *Devices, Circuits and Systems, 2004. Proceedings of the Fifth IEEE International Caracas Conference on*, vol. 1, 3-5 2004, pp. 315 – 318.
- [20] F. Microelectronics, “2008.1 product guide, assp, memory, asic,” 2008.
- [21] K. Takahashi and M. Sekiguchi, “Through silicon via and 3-d wafer/chip stacking technology,” in *VLSI Circuits, 2006. Digest of Technical Papers. 2006 Symposium on*, 0-0 2006, pp. 89 – 92.
- [22] J.-S. Kim, C. S. Oh, H. Lee, D. Lee, H.-R. Hwang, S. Hwang, B. Na, J. Moon, J.-G. Kim, H. Park, J.-W. Ryu, K. Park, S.-K. Kang, S.-Y. Kim, H. Kim, J.-M. Bang, H. Cho, M. Jang, C. Han, J.-B. Lee, K. Kyung, J.-S. Choi, and Y.-H. Jun, “A 1.2v 12.8gb/s 2gb mobile wide-i/o dram with 4 #x00d7;128 i/os using tsv-based stacking,” in *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, feb. 2011, pp. 496 –498.
- [23] J. Cong, J. Wei, and Y. Zhang, “A thermal-driven floorplanning algorithm for 3D ICS,” in *ICCAD '04: Proceedings of the 2004 IEEE/ACM International conference on Computer-aided design.* Washington, DC, USA: IEEE Computer Society, 2004, pp. 306–313.

- [24] J. Cong, A. Jagannathan, Y. Ma, G. Reinman, J. Wei, and Y. Zhang, "An automated design flow for 3d microarchitecture evaluation," in *Design Automation, 2006. Asia and South Pacific Conference on*, Jan 2006, pp. 6 pp.-.
- [25] Y. Xie, G. H. Loh, B. Black, and K. Bernstein, "Design space exploration for 3d architectures," *J. Emerg. Technol. Comput. Syst.*, vol. 2, no. 2, pp. 65–103, Apr. 2006. [Online]. Available: <http://doi.acm.org/10.1145/1148015.1148016>
- [26] Y. Xie and Y. Ma, "Design space exploration for 3d integrated circuits," in *Solid-State and Integrated-Circuit Technology, 2008. ICSICT 2008. 9th International Conference on*, oct. 2008, pp. 2317 –2320.
- [27] F. Robert, "Reconfigurable architectures," University Lecture, Université libre de Bruxelles, 2013.
- [28] S. Mohanty and V. K. Prasanna, "Rapid system-level performance evaluation and optimization for application mapping onto soc architectures," in *in Proc. of the IEEE International ASIC/SOC Conference*, 2002.
- [29] K. Ueda, K. Sakanushi, Y. Takeuchi, and M. Imai, "Architecture-level performance estimation method based on system-level profiling," *Computers and Digital Techniques, IEEE Proceedings -*, vol. 152, no. 1, pp. 12–19, 2005.
- [30] ESTECO. (2001) Esteco - leader in engineering design optimization software, process integration and multidisciplinary optimization with modefrontier @ONLINE. [Online]. Available: <http://www.esteco.com/index.jsp>
- [31] MULTICUBE. (2008) Multicube @ONLINE. [Online]. Available: <http://www.multicube.eu/>
- [32] C. Silvano, G. Palermo, V. Zaccaria, W. Fornaciari, R. Zafalon, S. Bocchio, M. Martinez, M. Wouters, G. Vanmeerbeeck, P. Avasare, L. Onesti, C. Kavka, U. Bondi, G. Mariani, E. Villar, H. Posadas, C. Y. Q., F. Dongrui, and Z. Hao, "Multicube: Multi-objective design space exploration of multiprocessor architectures for embedded multimedia applications," in *Proceedings of the DATE'09 workshop on Designing for Embedded Parallel Computing Platforms: Architectures, Design Tools, and Applications*, Nice, France, April 2009.
- [33] M. Explorer. (2009) Multicube explorer @ONLINE. [Online]. Available: http://home.dei.polimi.it/zaccaria/multicube_explorer_v1/Home.html
- [34] Multicube-SCoPE. (2009) Multicube-scope @ONLINE. [Online]. Available: <http://www.teisa.unican.es/gim/en/scope/multicube.html>
- [35] SCoPE. (2004) Scope @ONLINE. [Online]. Available: http://www.teisa.unican.es/gim/en/scope/scope_web/scope_home.php
- [36] C. Weis, N. Wehn, L. Igor, and L. Benini, "Design space exploration for 3d-stacked dram," in *Design, Automation Test in Europe Conference Exhibition (DATE), 2011*, march 2011, pp. 1 –6.

- [37] D. Milojevic, R. Radojcic, R. Carpenter, and P. Marchal, “Pathfinding: A design methodology for fast exploration and optimisation of 3d-stacked integrated circuits,” in *System-on-Chip, 2009. SOC 2009. International Symposium on*, oct. 2009, pp. 118–123.
- [38] D. Milojevic, T. Carlson, K. Croes, R. Radojcic, D. F. Ragett, D. Seynhaeve, F. Angiolini, G. V. der Plas, and P. Marchal, “Automated pathfinding tool chain for 3D-stacked integrated circuits: Practical case study.” in *3DIC*. IEEE, 2009, pp. 1–6.
- [39] G. B. Dantzig, *Maximization of a Linear Function of Variables Subject to Linear Inequalities, in Activity Analysis of Production and Allocation.* New York: Wiley, 1951, ch. XXI.
- [40] N. Karmarkar, “A new polynomial-time algorithm for linear programming,” *Combinatorica*, vol. 4, no. 4, pp. 373–395, Dec. 1984. [Online]. Available: <http://dx.doi.org/10.1007/BF02579150>
- [41] P. Vincke, *Multicriteria Decision-Aid.* J. Wiley, New York, 1992.
- [42] M. Ehrgott and X. Gandibleux, *Multiple Criteria Optimization. State of the art annotated bibliographic surveys.* Kluwer Academic, Dordrecht, 2002.
- [43] R. E. Steuer, *Multiple Criteria Optimization: Theory, Computation and Application.* John Wiley, New York, 546 pp, 1986.
- [44] J. Holland, “Adaptation in natural and artificial systems,” 1975.
- [45] F. Glover, “Heuristics for integer programming using surrogate constraints,” *Decision Sciences*, vol. 8, no. 1, pp. 156–166, 1977.
- [46] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, “Optimization by simulated annealing,” *Science*, vol. 220, pp. 671–680, 1983.
- [47] F. Glover, “Future paths for integer programming and links to artificial intelligence,” *Computers & Operations Research*, vol. 13, no. 5, pp. 533–549, 1986.
- [48] P. Moscato, “On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms,” Caltech Concurrent Computation Program, Tech. Rep., 1989.
- [49] M. Dorigo, “Optimization, Learning and Natural Algorithms (in Italian),” Ph.D. dissertation, Politecnico di Milano, Italy, 1992.
- [50] J. Dréo, A. Pétrowski, P. Siarry, and E. Taillard, *Metaheuristics for Hard Optimization.* Springer, 2006.
- [51] R. T. Marler and J. S. Arora, “Survey of multi-objective optimization methods for engineering,” *Structural and Multidisciplinary Optimization*, vol. 26, no. 6, pp. 369–395, Apr. 2004. [Online]. Available: <http://dx.doi.org/10.1007/s00158-003-0368-6>

- [52] K. Vekaria and C. Clack, “Selective crossover in genetic algorithms: An empirical study,” in *Parallel Problem Solving from Nature — PPSN V*, ser. Lecture Notes in Computer Science, A. Eiben, T. Bäck, M. Schoenauer, and H.-P. Schwefel, Eds. Springer Berlin Heidelberg, 1998, vol. 1498, pp. 438–447. [Online]. Available: <http://dx.doi.org/10.1007/BFb0056886>
- [53] S. Picek, M. Golub, and D. Jakobovic, “Evaluation of crossover operator performance in genetic algorithms with binary representation,” in *Bio-Inspired Computing and Applications*, ser. Lecture Notes in Computer Science, D.-S. Huang, Y. Gan, P. Premaratne, and K. Han, Eds. Springer Berlin Heidelberg, 2012, vol. 6840, pp. 223–230. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-24553-4_31
- [54] N. Srinivas and K. Deb, “Multiobjective optimization using nondominated sorting in genetic algorithms,” *Evolutionary Computation*, vol. 2, pp. 221–248, 1994.
- [55] E. Zitzler, L. Thiele, M. Laumanns, C. Fonseca, and V. da Fonseca, “Performance assessment of multiobjective optimizers: an analysis and review,” *Evolutionary Computation, IEEE Transactions on*, vol. 7, no. 2, pp. 117 – 132, april 2003.
- [56] V. Belton and T. Stewart, *Multiple Criteria Decision Analysis: An Integrated Approach*. Dordrecht: Kluwer Academic, 2002.
- [57] J. Dyer, “MAUT – multiattribute utility theory,” in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston, Dordrecht, London: Springer Verlag, 2005, pp. 265–285. [Online]. Available: <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>
- [58] T. Saaty, “The Analytic Hierarchy and Analytic Network Processes for the Measurement of Intangible Criteria and for Decision-Making,” in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston, Dordrecht, London: Springer Verlag, 2005, pp. 345–408. [Online]. Available: <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>
- [59] R. Benayoun, J. de Montgolfier, J. Tergny, and O. Larichev, “Linear programming with multiple objective functions: Step method (STEM),” *Mathematical Programming*, vol. 1, no. 3, pp. 366–375, 1971.
- [60] H. Nakayama and Y. Sawaragi, “Satisficing trade-off method for multiobjective programming,” in *Interactive Decision Analysis*, ser. Lecture Notes in Economics and Mathematical Systems, M. Grauer and A. Wierzbicki, Eds. Springer Berlin Heidelberg, 1984, vol. 229, pp. 113–122. [Online]. Available: http://dx.doi.org/10.1007/978-3-662-00184-4_13

- [61] J. Brans and P. Vincke, "A preference ranking organization method," *Management Science*, vol. 31, no. 6, pp. 647–656, 1985.
- [62] R. Benayoun, B. Roy, and B. Sussman, "ELECTRE: une méthode pour guider le choix en présence des points de vue multiples," SEMA-METRA International, Direction Scientifique, Tech. Rep., 1966, note de travail 49.
- [63] J. Figueira, S. Greco, and M. Ehrgott, *Multiple Criteria Decision Analysis: State of the Art Surveys*. Boston, Dordrecht, London: Springer Verlag, 2005.
- [64] P. Fishburn, *Utility Theory for Decision Making*. Wiley, New York, 1970.
- [65] R. Keeney and H. Raiffa, *Decisions with multiple objectives: Preferences and value tradeoffs*. J. Wiley, New York", 1976.
- [66] B. Mareschal and J. Brans, "Geometrical representations for MCDA," *European Journal of Operational Research*, vol. 34, pp. 69–77, 1988.
- [67] J. Brans, B. Mareschal, and P. Vincke, "PROMETHEE: a new family of outranking methods in multicriteria analysis," in *Operational Research, IFORS 84*, J. Brans, Ed. North Holland, Amsterdam, 1984, pp. 477–490.
- [68] M. Behzadian, R. Kazemzadeh, A. Albadvi, and M. Aghdasi, "PROMETHEE: A comprehensive literature review on methodologies and applications," *European Journal of Operational Research*, vol. 200, no. 1, pp. 198–215, 2010. [Online]. Available: <http://EconPapers.repec.org/RePEc:eee:ejores:v:200:y:2010:i:1:p:198-215>
- [69] J. Figueira, V. Mousseau, and B. Roy, "ELECTRE methods," in *Multiple Criteria Decision Analysis: State of the Art Surveys*, J. Figueira, S. Greco, and M. Ehrgott, Eds. Boston, Dordrecht, London: Springer Verlag, 2005, pp. 133–162. [Online]. Available: <http://www.springeronline.com/sgw/cda/frontpage/0,11855,5-165-22-34954528-0,00.html>
- [70] P. E. Black. (2006) Manhattan distance, in dictionary of algorithms and data structures @online. [Online]. Available: <http://xlinux.nist.gov/dads//HTML/manhattanDistance.html>
- [71] H. Yan, Q. Zhou, and X. Hong, "Thermal aware placement in 3D ICs using quadratic uniformity modeling approach," *Integr. VLSI J.*, vol. 42, no. 2, pp. 175–180, 2009.
- [72] G. Pelosi, "The finite-element method, part i: R. l. courant [historical corner]," *Antennas and Propagation Magazine, IEEE*, vol. 49, no. 2, pp. 180 –182, april 2007.
- [73] L. Davis, "Adapting operator probabilities in genetic algorithms," in *Proceedings of the third international conference on Genetic algorithms*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1989, pp. 61–69. [Online]. Available: <http://dl.acm.org/citation.cfm?id=93126.93146>

- [74] J. Hart and A. Shogan, “Semi-greedy heuristics: An empirical study,” *Operations Research Letters*, vol. 6, pp. 107–114, 1987.
- [75] F.-J. Veredas, M. Scheppeler, W. Moffat, and B. Mei, “Custom implementation of the coarse-grained reconfigurable adres architecture for multimedia purposes.” in *FPL*, T. Rissa, S. J. E. Wilton, and P. H. W. Leong, Eds. IEEE, 2005, pp. 106–111.
- [76] K. Lidouh, N. Doan, and Y. De Smet, “PROMETHEE-compatible presentations of multicriteria evaluation tables,” SMG, CoDE, Université Libre de Bruxelles, Brussels, Belgium, Tech. Rep. TR/SMG/2014-002, May 2014.
- [77] D. S. Vianna, J. E. C. Arroyo, P. S. Vieira, and T. R. de Azeredo, “Parallel strategies for a multi-criteria grasp algorithm.” in *SCCC*, 2005, pp. 116–122.
- [78] A. S. Abdallah, “Clock Based SoC Design, Towards a Design Space Exploration in MARTE,” Theses, Université des Sciences et Technologie de Lille - Lille I, Mar. 2011. [Online]. Available: <https://tel.archives-ouvertes.fr/tel-00597031>

Appendix

I Scalability case study - evaluation table

Alt.	Inter.	Cost	Vol.	Clk	Power
1	67.1437	345.4001	7.4892	21.94	127.3416
2	67.7864	345.4001	7.4892	21.94	122.2465
3	57.6018	378.6057	7.4892	21.94	125.0883
4	68.4043	345.4001	7.4892	21.94	122.3357
5	67.7316	345.4001	7.4892	21.94	123.0006
6	47.3105	331.4099	9.5167	29.64	109.4241
7	60.6876	334.7414	10.2556	24.32	115.2110
8	58.5433	380.2269	12.4531	28.24	98.5728
9	68.3753	321.4214	12.4531	14.72	102.6114
10	41.4362	393.2764	11.3974	12.94	116.2991
11	41.0517	332.5171	9.5362	29.32	116.8199
12	44.9002	352.2112	10.2724	19.12	117.6486
13	44.9635	327.5125	9.2938	14.76	150.0761
14	46.9319	372.3325	9.9838	10.44	108.0185
15	56.1703	379.5890	9.9838	13.14	98.8526
16	75.7641	347.6755	9.9838	22.14	106.6269
17	75.8782	347.6755	9.9838	22.14	107.0287
18	59.8297	348.3533	9.9838	15.64	102.8682
19	54.6413	400.4910	10.8763	11.54	100.5127
20	55.5439	327.4200	12.6941	15.56	127.7859
21	50.9166	317.2817	10.2605	20.86	118.8282
22	61.4915	322.8419	8.3174	29.34	142.2117
23	55.3600	361.6828	10.8977	31.76	96.8864
24	45.2119	417.1320	12.8873	24.24	95.5818
25	73.8553	326.8663	9.5001	16.36	109.8921
26	37.6445	320.7500	11.5028	11.86	157.4299
27	36.9018	323.2837	10.8478	11.66	146.0659
28	36.2394	338.2235	11.5028	10.16	151.4752
29	64.3923	418.2308	9.5458	27.06	100.6367
30	48.9768	354.5924	12.6941	26.82	100.5739
31	31.9275	431.8478	8.3639	32.34	120.5627
32	28.9445	368.8540	8.5959	32.34	121.0761
33	57.8361	409.9552	10.8658	30.46	97.1735
34	53.9757	441.6485	10.8658	30.46	98.4224
35	48.2477	412.2440	8.0452	19.54	125.5353

Alt.	Inter.	Cost	Vol.	Clk	Power
36	44.2677	490.4301	12.5707	14.24	98.9135
37	39.1444	396.6074	9.2943	14.22	105.9363
38	51.7699	391.0361	9.0224	14.72	102.4671
39	38.8748	395.2614	9.2943	13.42	127.6504
40	53.1014	511.7339	10.8658	30.46	97.2863
41	55.4079	440.3307	8.8679	21.64	109.1315
42	43.7142	350.4684	10.4325	29.64	106.8218
43	40.4749	412.8290	9.5167	29.26	105.3220
44	35.4249	473.6215	9.3517	26.82	101.1535
45	22.3437	460.7526	9.5167	18.84	114.1096
46	54.7345	439.9054	12.0184	31.76	98.8338
47	53.9511	439.9054	12.0184	33.96	94.2783
48	45.6106	334.1571	8.4218	14.76	151.9635
49	75.6979	556.7591	11.6992	7.54	108.0221
50	32.6110	338.8461	8.8063	33.76	145.1223
51	59.1314	529.5249	11.6992	9.24	103.7090
52	66.5220	414.0414	9.1875	5.64	127.1133
53	68.8931	346.8552	7.4892	21.74	121.3420
54	57.3688	390.0994	10.8977	31.76	95.7242
55	58.8415	390.0994	10.8977	31.76	96.0155
56	59.1594	390.0994	10.8977	31.76	96.0277
57	54.9439	353.4165	12.6941	26.82	99.0506
58	57.2414	422.7552	10.8977	31.76	96.2415
59	53.8370	346.1536	8.5959	29.54	121.4288
60	53.1831	298.7061	12.6941	22.06	114.3782
61	42.6019	354.8774	8.3049	21.84	126.9813
62	52.9383	478.4032	10.8977	31.76	97.7516
63	60.5928	379.5890	9.9838	13.14	99.7596
64	55.1757	312.3621	9.8092	29.64	108.4054
65	35.7660	305.1366	8.8081	14.76	152.3996
66	68.9482	358.5851	12.4531	14.22	102.3315
67	56.8187	342.8449	6.9513	21.96	140.5623
68	100.8476	426.3386	12.3678	19.46	94.2662
69	42.7333	429.7475	10.8658	19.96	99.1190
70	70.1955	344.7131	8.3890	19.46	119.3050
71	59.4849	494.0730	10.8658	28.66	97.3414
72	55.2610	390.8226	7.4892	23.02	123.5170
73	56.2332	454.8249	10.8658	30.46	96.8138
74	56.0679	468.1574	11.0691	30.46	96.4121
75	56.3206	454.8249	10.8658	30.46	96.9986
76	56.7347	454.8249	10.8658	30.46	96.4880
77	55.9827	454.8249	10.8658	30.46	97.7561
78	59.3294	454.8249	10.8658	30.46	96.3436
79	43.2587	483.1022	11.9658	30.46	97.7299
80	51.1697	318.0677	9.0813	35.76	113.2998
81	54.8029	414.8593	10.5993	34.24	97.4695
82	57.7455	287.8231	11.8119	32.66	121.0908
83	73.5559	438.7583	12.3678	20.86	96.3733
84	56.3632	306.3343	11.8993	32.36	118.7371
85	43.3298	454.7806	12.3678	19.46	99.1698

Alt.	Inter.	Cost	Vol.	Clk	Power
86	73.7186	291.7806	11.0260	19.46	111.8533
87	57.7091	360.2522	8.3890	19.46	117.4501
88	74.6318	329.2087	7.1983	17.16	128.2028
89	63.5617	346.8833	7.1983	31.46	130.9974
90	30.8621	400.3657	9.1355	31.96	119.2233
91	72.5163	372.0953	9.5167	29.64	105.8705
92	66.0537	369.4947	11.4923	19.66	98.6959
93	77.8276	452.7632	8.9558	19.46	109.2791
94	50.4464	490.4301	12.5707	14.24	98.5152
95	76.8637	318.0227	12.4531	12.74	100.7783
96	54.4188	440.3307	8.8679	21.64	109.2133
97	57.1054	278.3202	10.3593	23.74	133.5718
98	47.1377	496.8517	12.6941	21.74	94.5614
99	81.6104	381.1627	10.8658	29.36	94.9476
100	53.3402	486.1285	10.1350	34.74	97.6307
101	68.0443	377.4104	12.8873	24.24	97.7235
102	32.0455	324.9844	7.5432	28.96	155.2454
103	84.2261	428.1332	8.7660	34.74	102.7804
104	53.1241	486.1285	10.1350	32.24	98.3764
105	52.7889	486.1285	10.1350	34.64	98.4685
106	73.9236	404.6075	8.7660	34.74	105.3392
107	37.6601	306.4263	8.5959	32.34	120.8261
108	73.2667	357.4979	11.0944	22.36	99.2449
109	49.0023	353.3792	11.0944	22.36	101.8214
110	73.9457	353.3792	11.0944	22.36	100.2309
111	74.9185	356.8307	11.0944	22.36	99.1744
112	74.1909	353.3792	11.0944	22.36	99.9047
113	72.9825	403.7095	12.1961	22.36	96.5349
114	54.0551	318.1933	12.6941	18.64	121.2905
115	79.9484	270.9378	11.5017	31.46	121.1028
116	67.6957	381.5379	10.8658	30.46	96.0618
117	56.1407	381.5379	10.8658	30.46	98.2528
118	53.3194	388.2291	10.8658	30.46	99.6646
119	59.4928	380.2269	12.4531	27.64	96.1539
120	66.7415	381.5379	10.8658	30.46	97.2829
121	42.7157	412.8290	9.5167	29.64	104.9874
122	47.5413	444.2982	8.8224	22.04	113.5096
123	56.1213	512.4664	10.5993	30.86	97.0809
124	54.3092	288.4866	9.5167	27.96	118.7996
125	56.8597	305.1638	9.5167	27.96	115.7110
126	47.5859	398.4497	8.7746	19.46	124.3053
127	54.0043	408.4362	8.0280	28.54	109.7394
128	58.1685	404.2767	8.5625	31.14	112.0535
129	58.6016	345.8977	10.9950	36.74	103.7678
130	55.4697	345.8977	10.9950	38.94	98.1309
131	46.1720	286.3536	11.9046	37.86	125.1303
132	52.6912	400.0987	10.9950	38.94	99.5115
133	54.4271	394.0360	8.0280	29.64	118.5165
134	59.4737	392.6381	8.8679	24.44	115.3807
135	52.8188	394.0360	8.0280	30.94	121.2273

Alt.	Inter.	Cost	Vol.	Clk	Power
136	55.8758	298.9164	11.9046	29.76	118.5249
137	47.0295	395.1061	8.0212	29.64	118.5722
138	44.4510	472.4002	9.3450	34.74	98.8457
139	18.2582	437.2638	12.3765	19.64	133.0083
140	54.6047	410.2074	8.0280	27.96	120.0238
141	54.0908	408.4362	8.0280	28.54	109.9283
142	44.7667	417.1955	8.1788	33.74	113.7118
143	46.4718	408.4362	8.0280	28.54	112.7864
144	54.9063	408.4362	8.0280	28.54	110.1660
145	54.0817	408.4362	8.0280	28.54	110.5983
146	49.7030	408.4362	8.0280	33.04	114.5409
147	49.0215	444.5704	8.7601	31.14	112.2014
148	59.6176	356.8307	11.0944	19.26	108.8385
149	66.5703	341.6386	7.4892	20.44	129.0848
150	59.9975	521.9933	10.8658	27.96	96.0588
151	38.3503	383.9737	9.6007	37.96	114.5707
152	31.2267	400.1136	9.8126	31.96	106.4101
153	31.1118	400.3657	9.1355	31.96	118.8076
154	47.1843	435.0963	9.3450	34.74	101.0556
155	34.9439	400.3657	9.1355	31.96	116.7625
156	31.3780	400.3657	9.1355	31.96	116.9787
157	30.9287	407.2777	9.1355	31.96	114.3962
158	45.2024	489.4444	10.5993	30.86	99.3862
159	48.4691	375.1244	7.6305	14.76	149.3286
160	42.1125	346.2995	11.2707	22.36	104.6279
161	45.8106	343.3617	11.0944	22.36	103.7713
162	44.4114	330.3757	11.5017	31.46	122.1135
163	43.2611	355.5664	9.2938	14.76	150.5290
164	50.8215	303.8639	8.4456	19.46	122.2705
165	44.4680	295.2336	11.5017	31.46	121.4953
166	56.2292	351.4315	10.7705	22.56	100.1391
167	72.9891	311.8305	12.6941	31.46	103.9234
168	67.8289	277.8022	12.6941	31.46	120.6805
169	35.8265	309.0332	7.1941	29.36	149.3435
170	47.0899	359.6294	10.7117	27.24	109.1292
171	52.5234	325.4054	8.7134	35.66	116.3160
172	65.6790	424.6716	9.0224	14.24	99.0998
173	46.9313	484.6466	10.9298	14.22	105.3430
174	49.9400	359.7155	9.0224	14.22	107.4596
175	66.6063	351.1630	7.4892	23.02	127.4326
176	38.7615	364.7985	8.2615	36.64	139.5135
177	85.2155	316.1776	12.5620	15.86	110.8522
178	93.8050	364.4797	8.1682	25.74	115.5890
179	78.8251	303.1227	12.5620	19.36	119.4254
180	78.9629	303.1227	12.5620	19.36	119.8466
181	37.5168	414.2434	9.8345	15.84	127.7761
182	32.5520	399.2970	11.0456	15.86	123.4950
183	37.2693	350.9679	8.3522	33.76	143.5792
184	36.4096	395.3331	11.0456	15.76	122.0380
185	22.7388	373.1171	7.4956	22.44	121.7426

Alt.	Inter.	Cost	Vol.	Clk	Power
186	34.1353	393.6141	11.0456	15.86	128.8091
187	34.9099	307.2429	9.2943	15.86	135.5480
188	36.0948	396.6303	12.3004	15.86	122.6954
189	72.0929	288.1649	12.7738	21.86	121.1848
190	46.4876	340.2722	9.5167	29.64	108.0806
191	56.3674	512.4664	10.5993	30.86	94.9808
192	65.4383	393.3299	10.5993	34.24	94.8932
193	37.5239	348.8331	10.8027	32.34	121.0732
194	60.8142	408.7385	10.8658	30.46	97.2360
195	36.7935	346.6650	10.8027	32.34	121.5763
196	52.6566	545.0644	12.5707	14.24	98.2065
197	79.6736	352.9201	9.5167	29.64	106.0659
198	59.2152	390.3681	10.8783	29.66	98.6472
199	66.6735	351.1630	7.4892	23.02	126.0170
200	66.4671	351.1630	7.4892	23.12	127.3275
201	73.9825	288.1649	12.7738	21.86	119.8945
202	60.8182	476.3486	10.8783	29.66	97.3634
203	50.6481	334.3903	9.2525	35.76	112.0129
204	47.6071	350.0589	9.7727	13.52	112.6410
205	70.2186	317.8574	12.4531	19.32	108.2199
206	55.0022	448.5569	9.9838	11.84	105.9511
207	75.1913	318.0227	12.4531	13.84	95.4934
208	90.0382	288.1649	12.7738	19.46	122.8937
209	58.6635	380.2269	12.4531	28.24	97.6012
210	71.7585	365.4396	12.4531	14.24	97.9771
211	71.0826	321.4214	12.4531	14.72	101.4576
212	74.1808	332.5934	12.4531	13.02	99.8575
213	70.2325	417.6606	11.1020	29.66	93.3132
214	49.3959	328.8576	8.9205	35.76	112.2172
215	29.0077	552.8741	12.0067	9.94	113.4470
216	64.9424	350.1117	7.5215	23.12	129.2964
217	43.3768	345.0730	8.0212	26.04	139.0607
218	38.5967	359.4228	8.9439	26.04	138.2159
219	36.8103	338.2235	11.5028	10.16	148.4535
220	50.1248	319.5091	9.3477	32.34	120.5121
221	54.9956	234.2890	11.5017	31.46	122.0863
222	49.9631	419.6632	9.3450	34.74	100.6159
223	43.3631	405.8584	9.3450	34.74	102.9874
224	63.7354	400.1639	12.8873	24.24	96.6318
225	42.7329	410.0575	8.4405	20.04	131.6816
226	51.6068	364.1128	8.0452	32.14	116.5314
227	42.0865	516.4367	11.1916	34.74	97.7231
228	45.3650	516.1457	12.4531	14.22	104.1458
229	61.9402	337.5795	8.0212	28.44	131.2622
230	51.6487	327.8256	8.8586	35.56	119.4363
231	56.7730	363.7903	11.0944	22.36	99.7696
232	48.2236	360.8082	8.5959	25.24	118.9967
233	51.5201	424.4922	11.8506	32.94	97.7779
234	79.3979	403.7095	12.1961	19.96	97.8826
235	50.5058	347.8662	9.2803	22.36	118.8251

Alt.	Inter.	Cost	Vol.	Clk	Power
236	52.5607	454.2770	8.9558	22.36	108.8546
237	52.7399	454.2770	8.9558	22.36	107.4342
238	42.9127	412.8290	9.5167	29.64	104.9763
239	43.7190	510.3282	11.5870	34.74	97.5286
240	70.2453	381.5379	10.8658	26.76	98.6151
241	61.9978	413.7253	10.8658	26.76	93.8789
242	48.5015	325.7272	9.8407	19.46	119.1958
243	50.2768	437.6252	10.4227	34.74	98.3386
244	33.3719	339.9392	7.5432	28.96	154.8905
245	53.7598	417.1321	10.8658	30.46	98.9397
246	78.4504	381.1627	10.8658	28.96	97.3324
247	81.9728	381.1627	10.8658	30.46	97.8597
248	65.2511	394.4953	11.0691	30.46	98.1996
249	84.3657	381.1627	10.8658	30.46	97.1864
250	81.8917	437.4766	10.8658	24.96	98.0183
251	47.2260	339.1342	9.8407	19.46	119.9051
252	35.2501	388.8749	10.8478	9.94	128.1933
253	56.3112	486.1285	10.1350	34.74	97.3521
254	68.3044	332.5934	12.4531	14.22	104.8495
255	26.7757	409.3761	9.2943	22.44	117.1989
256	72.6471	414.2575	9.3450	34.74	101.3048
257	61.2232	336.9966	8.1036	36.56	122.7024
258	52.8166	350.6923	10.1392	13.42	124.7007
259	46.1694	377.7403	9.0224	14.22	106.2996
260	37.1649	328.0752	9.2943	10.14	139.0130
261	56.8749	360.6774	8.2782	18.14	133.8666
262	53.5431	488.7379	10.2916	34.74	97.1421
263	58.3334	360.6774	8.2782	18.14	135.0653
264	37.7243	412.2124	9.2943	14.22	107.8078
265	50.5740	337.1089	9.2943	18.14	131.5942
266	38.6168	304.2286	11.5028	10.14	150.7673
267	47.1505	401.0309	9.2943	21.26	105.8057
268	41.5383	355.1282	9.2943	23.52	123.5744
269	32.2964	447.5980	9.4822	12.64	114.6895
270	49.6060	316.8820	8.8537	14.76	150.3118
271	37.6978	540.2523	12.0067	9.94	119.2606
272	32.0038	418.2046	8.3639	25.24	117.4724
273	28.9176	446.9796	9.2943	32.34	107.6907
274	33.7896	423.4145	9.0873	11.34	116.6936
275	28.4046	409.3761	9.2943	22.44	114.7682
276	44.6205	486.1285	10.1350	32.64	98.8703
277	51.6525	353.7626	11.6327	22.36	100.9465
278	53.7042	373.6541	11.0944	22.36	99.2147
279	57.8447	402.3695	12.4531	28.24	98.2853
280	49.3257	328.8576	8.9205	35.76	114.1049
281	54.0690	511.7339	10.8658	30.46	94.5515
282	49.2827	315.3877	8.9205	35.76	115.3900
283	54.3315	511.7339	10.8658	30.46	96.3061
284	58.2386	392.6381	8.8679	25.44	113.3499
285	58.9254	317.6194	9.2118	27.84	115.7758

Alt.	Inter.	Cost	Vol.	Clk	Power
286	44.3364	386.9664	8.3287	18.64	127.7590
287	56.1186	353.2042	8.0212	24.74	137.0858
288	44.0197	407.8024	8.3287	13.56	142.6864
289	32.6393	408.6840	8.3756	14.76	145.7999
290	32.2974	383.8006	8.0212	21.04	130.3954
291	33.5330	407.9252	9.4686	16.34	129.7503
292	44.6006	334.3129	7.0321	26.04	140.1545
293	35.7913	330.0914	7.9878	37.34	128.0082
294	61.6874	466.0088	10.3691	17.64	96.1573
295	62.8978	466.0088	10.3691	17.64	95.6477
296	53.3307	440.6843	11.1718	17.64	96.2172
297	21.9975	480.3911	12.3765	13.56	148.6316
298	51.6770	327.2837	8.7134	36.56	118.4748
299	52.4152	407.8024	8.3287	13.56	137.6340
300	30.1791	407.8024	8.3287	13.56	146.0116
301	49.2599	479.3314	9.9838	10.36	114.7688
302	51.9531	407.8024	8.3287	13.56	143.2845
303	67.7175	346.7237	12.4531	17.32	105.6336
304	51.5886	407.8024	8.3287	13.56	143.8056
305	23.5363	480.3911	12.3765	13.56	143.7344
306	19.6694	411.2197	12.3765	21.04	134.3587
307	42.8300	362.2327	9.5167	29.64	112.5624
308	20.9050	411.2197	12.3765	16.34	135.4180
309	39.5801	361.7541	8.9439	26.04	138.0216
310	21.8322	411.2197	12.3765	19.34	133.8507
311	55.7786	353.3792	11.0944	22.36	101.2293
312	54.9422	337.5795	8.0212	26.04	136.5539
313	44.2646	345.0730	8.0212	26.04	139.1988
314	56.1210	343.7280	8.0212	26.04	130.3497
315	62.8885	337.5795	8.0212	28.74	129.7067
316	54.6815	337.5795	8.0212	26.04	136.8615
317	32.3145	367.8312	8.3522	33.76	144.6403
318	37.4743	359.4228	8.9439	26.04	139.5414
319	56.2538	337.5795	8.0212	26.04	136.9815
320	50.7434	407.8024	8.3287	13.56	144.0398
321	57.0332	337.5795	8.0212	26.04	136.5573
322	54.9193	355.2317	8.0212	27.34	132.2284
323	49.9045	355.2311	9.5167	29.64	107.7836
324	34.6224	542.6298	12.0067	9.94	119.4686
325	36.7945	413.6859	8.4405	27.24	117.2627
326	49.4347	315.3877	8.9205	35.76	113.3828
327	66.2821	393.3299	10.5993	34.24	95.2296
328	25.7009	455.1565	10.4039	21.44	117.5937
329	63.3406	313.9855	8.0438	22.44	121.1688
330	63.3488	312.3561	8.0438	22.44	120.3739
331	63.2559	301.3468	7.8742	22.44	121.1726
332	68.5387	332.5934	12.4531	14.22	103.0227
333	48.7472	339.2927	8.9205	35.76	112.6812
334	44.5686	389.3634	9.5167	29.64	105.9675
335	57.8955	297.1446	12.6941	17.34	121.7916

Alt.	Inter.	Cost	Vol.	Clk	Power
336	59.7430	456.5192	10.5993	30.44	96.1866
337	57.8304	337.7437	9.9838	14.22	112.2992
338	56.0284	410.7025	9.0224	14.22	103.4741
339	43.6173	457.7251	11.5322	9.24	108.3934
340	63.9349	332.4880	9.6685	14.22	112.4470
341	64.0690	325.3707	8.0438	21.44	122.3796
342	79.7881	420.1399	12.3130	34.24	92.8326
343	70.2180	398.2975	12.4531	14.24	96.6156
344	84.9141	429.5357	12.4531	7.34	124.5053
345	48.4829	328.8576	8.9205	35.76	114.1635
346	73.4357	434.1700	10.5993	30.44	93.4126
347	54.9181	241.4400	11.5017	31.46	123.3593
348	43.2682	345.1816	8.4218	7.34	129.0506
349	33.4643	448.9066	10.1402	7.34	126.7922
350	50.4974	343.7664	8.4218	7.34	132.3931
351	51.5279	343.7664	8.4218	7.34	131.4255
352	44.7419	406.7954	9.6907	7.34	126.2983
353	48.4745	400.0481	10.1578	7.64	127.4349
354	59.8442	388.8936	8.6544	14.22	115.1258
355	70.5626	317.8574	12.4531	19.32	109.0422
356	71.4675	333.8032	12.4531	14.74	99.3730
357	81.5133	288.1649	12.7738	19.46	123.0135
358	64.8714	397.7172	10.5993	30.94	94.8374
359	69.8532	322.9213	12.4531	25.14	97.9748
360	84.8058	302.0407	12.4531	9.94	115.3341
361	73.6783	342.4534	12.4531	13.62	101.5747
362	68.0735	334.3991	12.4531	14.72	101.7782
363	68.4998	409.4378	12.4531	14.74	97.3097
364	71.2620	321.4214	12.4531	14.72	102.5375
365	68.4850	321.4214	12.4531	14.72	104.4712
366	53.0006	517.2515	12.5707	14.24	98.4847
367	50.7338	322.9566	8.9205	35.76	112.9490
368	73.9038	350.7014	12.4531	13.32	103.6675
369	37.8533	396.6074	9.2943	14.22	106.7442
370	44.9851	491.0462	12.4531	14.22	105.2062
371	58.8865	355.1136	6.7809	14.22	116.0239
372	55.7258	508.7742	12.4531	13.84	98.2211
373	52.2775	472.3221	12.5707	14.24	98.9955
374	69.0014	347.4131	12.4531	14.72	101.1222
375	65.6283	395.5135	10.5993	30.44	96.1662
376	57.0354	371.3910	7.6042	14.22	115.6544
377	43.4537	293.3426	11.8119	32.66	119.7962
378	47.0173	393.6284	9.7727	7.34	128.3987
379	68.6512	358.5851	12.4531	14.22	102.9247
380	69.7614	332.5934	12.4531	14.22	103.0067
381	68.3655	332.5934	12.4531	14.22	105.8114
382	42.1082	410.0860	9.0224	14.24	109.3076
383	48.7162	421.7012	9.0224	14.24	100.0172
384	72.6554	408.6512	9.3422	14.24	101.1318
385	72.1025	365.4396	12.4531	14.24	98.5626

Alt.	Inter.	Cost	Vol.	Clk	Power
386	73.4242	434.1700	10.5993	30.44	93.7013
387	59.1728	502.6488	11.7361	14.24	97.1268
388	50.5444	521.8786	12.5707	14.24	98.3188
389	51.7347	490.4301	12.5707	14.24	98.9019
390	39.4079	360.5756	9.9308	14.24	100.0786
391	40.2490	436.6221	9.3422	14.24	99.5021
392	43.6432	480.3343	11.2658	14.24	99.0197
393	64.8127	465.9077	10.7443	14.24	98.8194
394	74.9306	380.2269	12.4531	21.34	96.1511
395	73.6792	316.0717	12.4531	23.44	92.5792
396	49.7389	403.2405	9.0224	25.84	104.7437
397	59.2490	380.2269	12.4531	28.24	98.0530
398	77.5178	341.8314	12.3130	34.24	92.8854
399	49.9310	416.0725	9.0224	28.24	102.1937
400	59.0928	386.8019	12.4531	28.24	97.2811
401	87.9475	393.9995	12.3130	30.74	91.8094
402	36.5647	374.2268	9.9722	24.84	114.6037
403	35.9105	339.0144	6.7011	29.36	146.7685
404	33.7155	330.8138	9.9722	24.82	127.4715
405	46.9882	349.5208	9.9722	24.82	126.1003
406	37.2534	362.7086	9.9722	24.82	124.2838
407	53.3825	384.9258	9.9838	11.84	106.2895
408	53.3324	394.2737	11.4071	34.24	99.1394
409	56.0230	408.8624	9.0224	14.22	104.8237
410	93.0125	450.2296	10.3166	34.74	96.6861
411	51.8419	376.1328	11.7681	14.22	106.5094
412	55.0019	383.0027	11.7681	14.22	104.4783
413	38.6559	415.4503	8.7640	20.04	130.8444
414	55.9826	447.1718	8.9097	15.74	121.3628
415	55.0009	334.5879	11.2802	16.36	109.7521
416	49.0667	426.3825	8.5062	19.54	123.4421
417	56.3659	410.2663	12.6941	17.44	93.4064
418	56.1041	412.8745	8.0830	19.54	122.8547
419	41.7444	437.9529	8.8679	21.64	109.8065
420	75.8204	335.2991	10.5993	34.24	98.4721
421	34.4772	469.6168	9.4822	12.64	109.3375
422	37.6900	447.5353	11.0277	5.24	157.3910
423	68.0738	381.5379	10.8658	30.46	98.5808
424	49.7067	427.3330	11.8506	32.94	99.9784
425	58.8076	317.6790	8.5959	32.34	120.5235
426	64.4960	398.0466	11.8564	30.16	97.1812
427	37.0510	352.4649	10.4300	19.62	115.8346
428	53.3150	419.6632	9.3450	34.74	98.1692
429	73.0544	425.4128	9.3450	34.74	97.6094
430	64.5068	398.0466	11.8564	30.16	97.2368
431	53.2752	504.0905	10.4227	34.74	97.6031
432	66.6069	351.1630	7.4892	23.02	126.8817
433	74.9058	288.1649	12.7738	21.86	120.8201
434	76.9333	288.1649	12.7738	21.86	119.9245
435	74.9763	288.1649	12.7738	21.86	119.9828

Alt.	Inter.	Cost	Vol.	Clk	Power
436	80.9047	288.1649	12.7738	19.46	123.1572
437	71.0137	288.1649	12.7738	19.46	123.2382
438	31.7027	362.7757	8.4654	28.96	154.7654
439	35.4008	406.4844	10.4039	24.66	118.6520
440	53.8935	325.4054	8.7134	35.76	115.9780
441	71.6603	289.8389	12.7738	20.96	122.8548
442	65.8589	351.1630	7.4892	23.02	129.2212
443	53.8338	317.8364	9.0675	34.96	111.0491
444	65.6525	351.1630	7.4892	23.12	130.4470
445	31.9117	422.6460	8.4405	20.04	131.0322
446	26.8908	376.7087	8.8081	12.46	146.5505
447	38.6646	351.2251	10.4300	19.62	116.0918
448	59.0153	458.2684	10.5993	30.94	93.9718
449	42.3341	373.6472	9.9838	12.52	116.4631
450	42.2839	358.0734	9.9838	12.52	116.9356
451	30.8280	336.7191	8.8063	36.56	148.9218
452	34.5546	331.2774	8.3522	33.76	145.2555
453	37.6940	351.3217	10.4300	19.62	117.1128
454	50.8649	318.0941	8.4405	27.22	126.5147
455	55.2634	353.3792	11.0944	22.36	101.3256
456	55.0461	374.8011	8.7601	27.22	114.7700
457	42.3874	410.0575	8.4405	20.04	133.3325
458	43.0935	412.8290	9.5167	28.34	105.0655
459	59.1498	335.2991	10.5993	34.24	100.9554
460	57.1578	475.6265	10.5993	30.86	98.2114
461	68.3635	406.8858	12.8873	18.14	97.2454
462	48.7044	353.4165	12.6941	25.72	103.0755
463	59.8521	401.6576	10.8658	32.66	97.4902
464	60.4730	452.1425	10.8658	30.16	97.6241
465	46.9499	461.6319	9.2299	11.34	114.1623
466	75.5352	318.0227	12.4531	13.84	95.7951
467	88.8882	427.9939	12.5620	11.34	104.8597
468	30.1468	461.6319	9.2299	11.34	115.0335
469	50.4109	512.3729	12.2744	8.94	116.0127
470	44.2186	461.6319	9.2299	11.34	114.5057
471	43.5169	466.4352	9.2299	11.34	113.8841
472	79.5653	500.8236	11.0508	11.34	102.6535
473	33.1922	502.3554	11.2658	11.34	113.1891
474	44.3301	461.6319	9.2299	11.34	115.4892
475	33.7614	502.1459	10.8713	11.34	112.9257
476	46.7912	461.6319	9.2299	11.34	114.3664
477	44.8472	461.6319	9.2299	11.34	116.4482
478	46.5836	501.9656	9.9237	11.34	113.7738
479	44.5784	393.7284	9.9838	11.34	108.9966
480	50.7045	367.8353	8.4405	27.22	122.4414
481	42.4280	410.0575	8.4405	20.04	133.3609
482	42.2169	410.0575	8.4405	21.84	129.7894
483	44.8059	410.0575	8.4405	22.24	126.0119
484	64.2939	363.5648	8.3049	27.14	111.7001
485	50.3384	353.4165	12.6941	26.82	100.8515

Alt.	Inter.	Cost	Vol.	Clk	Power
486	72.8200	401.2772	12.8873	24.24	96.1651
487	32.3805	339.0964	8.8063	33.76	147.5204
488	57.6658	371.1518	8.7601	27.14	110.7633
489	59.1766	407.8470	8.4405	32.24	115.4815
490	50.9014	353.4165	12.6941	26.82	100.4446
491	50.5859	360.8082	8.5959	25.24	119.2431
492	66.3297	351.1630	7.4892	23.12	130.4029
493	64.3218	353.9508	8.2257	21.64	121.7370
494	45.8342	318.0494	11.4776	32.66	118.8256
495	53.3343	308.2223	8.8537	14.76	151.6800
496	76.1330	386.9564	9.3450	34.74	104.4707
497	53.7629	463.0934	9.3450	34.74	97.4671
498	33.9409	353.6098	7.9155	28.96	154.4713
499	56.3187	486.1285	10.1350	34.74	96.8275
500	61.0329	428.2966	10.5993	34.24	96.0230
501	51.6523	325.4054	8.7134	36.06	120.6945
502	33.7373	529.2382	10.9121	26.82	100.6299
503	57.6113	353.3792	11.0944	16.26	109.7944
504	45.5379	325.2446	9.3477	32.34	120.8127
505	72.7069	403.7095	12.1961	22.36	98.1292
506	53.0439	519.1078	10.8658	30.46	96.8146
507	46.9276	491.8034	10.8658	30.46	98.7299
508	47.1747	502.9835	10.8658	30.46	97.9699
509	36.5920	457.7251	11.5322	10.54	111.6804
510	48.8638	478.7341	10.8658	30.46	97.7645
511	58.6960	524.4500	10.8658	24.96	95.2498
512	75.0791	529.3399	13.2443	12.96	100.1006
513	61.8664	413.7253	10.8658	30.46	97.0754
514	75.5716	510.9255	10.2692	24.96	97.9834
515	61.2163	368.0134	10.2692	32.86	99.7046
516	73.5768	403.7095	12.1961	22.36	98.4524
517	62.8510	368.0134	10.2692	32.06	98.9364
518	91.7994	401.3056	10.8658	26.76	96.9162
519	52.6258	355.7281	9.3630	32.34	116.8144
520	56.2427	394.8704	11.0691	30.46	97.5216
521	48.9545	353.4165	12.6941	26.82	101.7255
522	66.3269	390.7955	11.8564	30.46	97.8930
523	57.3509	452.1425	10.8658	30.46	98.0129
524	70.1478	381.5379	10.8658	30.46	98.3796
525	68.0223	381.5379	10.8658	30.46	98.7050
526	44.1669	299.9896	10.2605	26.44	142.9024
527	55.4858	283.0568	10.2605	28.22	130.6553
528	58.5379	410.9787	8.8679	25.44	112.2867
529	33.8012	365.7349	10.2023	23.76	136.7379
530	53.4753	486.1285	10.1350	34.74	97.5205
531	43.5842	486.1285	10.1350	34.74	97.8135
532	34.2669	309.0332	7.1941	29.36	153.0377
533	66.5146	393.3299	10.5993	34.24	95.8462
534	72.7773	406.0620	8.3104	16.36	111.6497
535	48.2457	330.6815	8.4218	14.76	149.7666

Alt.	Inter.	Cost	Vol.	Clk	Power
536	36.8893	363.1051	8.3522	33.76	142.9625
537	50.3205	428.6718	9.3450	34.74	98.2386
538	50.2746	325.4054	8.7134	36.56	120.2313
539	58.6787	308.2254	8.5959	32.34	120.7590
540	53.2016	398.9839	11.8506	32.94	95.5060
541	51.9039	288.0398	8.9205	35.76	114.7756
542	60.6189	335.2991	10.5993	34.24	99.0934
543	56.5276	464.7965	11.0348	13.76	97.5737
544	57.5044	390.2340	12.6941	27.74	95.0669
545	56.1225	398.9839	11.8506	32.94	95.8517
546	46.9308	344.3972	10.4089	22.36	111.9079
547	62.3480	313.4851	8.4420	16.36	111.3098
548	52.8866	459.0496	9.4822	12.64	110.0064
549	72.5679	281.3926	12.6941	25.76	113.2012
550	47.5454	341.5761	10.2397	22.06	111.2683
551	63.0052	312.3561	8.0438	22.44	122.8175
552	26.5962	368.4892	7.2179	22.44	126.6519
553	52.0171	430.3685	8.9174	22.44	116.8651
554	27.1704	403.4441	8.6553	22.44	119.9245
555	50.0686	360.8082	8.5959	25.24	119.5182
556	50.1189	406.9105	9.9838	13.14	101.5971
557	72.8044	326.8663	9.5001	16.36	109.9693
558	61.6785	414.0348	10.7443	12.94	102.5679
559	53.5319	435.1597	11.3172	11.84	105.5162
560	50.1966	406.9105	9.9838	13.14	102.5766
561	59.2793	363.4266	8.7601	28.34	107.2131
562	49.6371	444.3290	11.8673	10.14	102.0536
563	50.4916	353.3792	11.0944	22.36	103.1938
564	43.6939	429.2324	9.9838	12.52	115.3060
565	50.0376	491.4436	10.8763	11.54	101.6546
566	72.3954	347.6755	9.9838	22.14	107.5533
567	37.2647	370.9243	8.1715	25.24	116.8248
568	63.4940	341.1850	9.1875	13.06	151.9936
569	52.9410	301.0238	8.4218	14.76	152.2772
570	57.1076	356.1608	6.8365	31.14	138.6370
571	60.1939	300.1646	9.1875	14.76	156.1581
572	57.1037	356.1608	6.8365	31.14	139.8760
573	36.6714	365.2970	10.1275	25.32	123.1515
574	35.9148	309.0332	7.1941	29.36	146.6663
575	63.6705	489.6547	10.5993	30.86	92.2799
576	58.6356	353.3792	11.0944	16.26	109.6950
577	38.7901	332.6968	9.8345	32.34	117.3426
578	53.0684	315.3877	8.9205	36.96	113.2393
579	56.9749	295.4452	12.6941	17.76	114.9997
580	56.9755	347.9667	8.5959	25.24	118.0309
581	47.8863	360.8082	8.5959	25.24	119.6464
582	50.6728	360.8082	8.5959	25.24	118.4299
583	50.0779	319.5091	9.3477	32.14	123.2682
584	54.8325	367.4974	10.5993	34.24	96.8875
585	63.9499	300.3001	8.5959	32.34	119.9151

Alt.	Inter.	Cost	Vol.	Clk	Power
586	40.2109	332.6968	9.8345	32.34	119.6849
587	41.6710	408.7193	9.5167	29.64	105.1891
588	44.9674	357.7173	8.5959	26.74	120.6119
589	61.1907	489.6547	10.5993	30.86	94.7778
590	63.2187	340.1148	11.0944	16.26	111.8013
591	59.1344	353.3792	11.0944	22.36	100.4181
592	53.7963	437.9529	8.8679	21.64	109.4159
593	58.4573	392.6381	8.8679	25.44	112.4993
594	58.8978	319.1813	8.5959	32.34	112.0658
595	48.4399	315.3877	8.9205	35.76	115.5547
596	41.9290	398.7839	9.5167	29.64	105.6455
597	95.8922	440.1911	8.5397	16.36	109.4834
598	59.4944	334.0015	8.7134	32.04	116.9434
599	54.3675	501.4645	10.2651	34.74	97.3942
600	51.3691	434.9992	10.2651	34.74	98.0978
601	44.4764	501.4645	10.2651	34.74	97.6846
602	45.2800	358.7785	8.2615	34.44	129.7147
603	58.4944	356.8307	11.0944	16.26	109.5854
604	68.5718	379.4636	12.8873	24.24	97.4036
605	58.6569	392.6381	8.8679	25.44	112.3309
606	51.0076	358.7785	8.2615	34.44	132.2696
607	81.5134	316.1776	12.5620	15.86	124.2042
608	49.9488	425.8502	8.9205	35.76	111.6436
609	32.9149	367.8312	8.3522	33.76	142.3198
610	43.5318	412.8290	9.5167	28.34	105.1433
611	46.1256	291.7519	11.9046	37.86	124.4469
612	46.5393	255.8331	11.9046	37.86	123.1267
613	68.4675	341.9880	12.4531	14.22	103.3601
614	43.8684	322.7235	11.9046	34.96	114.0852
615	73.8278	356.3942	11.8650	34.24	98.2390
616	35.8315	309.0332	7.1941	29.36	149.2480
617	57.9193	353.3792	11.0944	16.26	110.4335
618	38.7568	369.2305	9.8510	27.06	118.9838
619	53.4858	437.9529	8.8679	21.64	110.2868
620	38.7654	369.2305	9.8510	26.66	118.2272
621	44.4191	364.7985	8.2615	36.74	133.1701
622	46.7813	339.0010	8.2615	35.14	116.1644
623	93.7921	366.1777	8.1682	25.74	115.8327
624	59.8868	347.5764	7.0409	33.76	125.7000
625	73.0461	409.1854	12.8873	24.24	95.9633
626	59.6599	400.2242	12.8873	24.24	97.0832
627	35.6782	407.9537	10.4039	22.36	116.7466
628	33.1304	342.4533	8.8081	14.76	154.1040
629	55.0449	535.3791	10.5993	30.86	98.0451
630	36.7698	367.6698	10.1566	14.22	107.8936
631	36.5781	304.2199	7.1941	29.36	149.6232
632	53.7299	486.1285	10.1350	34.74	97.8328
633	37.2711	352.8506	8.3522	33.76	143.5246
634	38.2722	309.0332	7.1941	30.36	143.9398
635	58.5409	353.3792	11.0944	22.36	100.9504

Alt.	Inter.	Cost	Vol.	Clk	Power
636	52.6672	396.0753	8.8679	30.84	115.9485
637	57.1257	337.5795	8.0212	26.04	136.7026
638	65.3724	346.6755	9.5167	29.64	106.4067
639	60.6337	392.6381	8.8679	25.44	112.3477
640	50.0660	333.8147	11.0944	22.36	104.4521
641	51.7929	333.8147	11.0944	22.36	105.3469
642	59.8595	353.3792	11.0944	22.36	100.6259
643	63.2506	379.1456	10.4298	26.82	99.0573
644	61.6056	380.9042	9.7467	26.82	103.4307
645	51.3474	353.4165	12.6941	26.82	101.0594
646	53.7684	326.1421	12.6941	15.86	114.8956
647	72.0525	225.4472	11.5017	31.46	125.0113
648	56.8868	347.3809	12.6941	15.42	113.3106
649	46.6795	260.3643	11.5017	31.46	124.7252
650	56.0974	486.1285	10.1350	34.74	97.4043
651	48.3296	383.7366	7.9757	37.34	120.3309
652	79.5129	348.5802	11.9583	32.94	96.0009
653	47.2531	444.3081	12.4531	14.22	105.6576
654	35.1866	354.7190	11.5028	10.16	151.8770
655	42.4662	388.1258	9.5167	29.64	107.7449
656	38.5070	351.2906	11.1910	23.52	134.6938
657	39.0985	453.3573	9.4822	17.66	104.5843
658	44.0531	386.4611	11.5322	13.62	114.3458
659	43.8890	457.7251	11.5322	13.04	109.5718
660	39.8895	418.9921	9.5167	25.84	103.1884
661	76.8630	525.9761	12.0067	8.04	111.0569
662	50.0812	315.3877	8.9205	35.76	113.5796
663	33.6335	324.9844	7.5432	28.96	154.5452
664	82.9815	362.0782	11.0944	21.84	99.3811
665	41.8590	329.7656	7.3652	37.34	121.3313
666	33.9918	367.8312	8.3522	33.76	141.9913
667	81.2773	437.9348	12.4149	22.04	95.8135
668	62.7308	366.0046	11.9583	32.94	96.2572
669	32.6930	336.7191	8.8063	28.76	136.4987
670	45.1595	375.5971	9.0548	22.36	126.5716
671	84.2425	428.1332	8.7660	34.74	102.4702
672	74.4557	333.4190	9.5001	16.36	109.3690
673	56.6882	353.3792	11.0944	16.26	111.4162
674	48.3758	441.0565	11.0944	16.26	89.9020
675	57.0944	357.6043	11.0944	16.26	110.0076
676	56.6395	353.3792	11.0944	16.26	111.6049
677	56.9450	353.3792	11.0944	16.26	111.9996
678	55.9434	353.3792	11.0944	22.36	101.1181
679	54.4965	322.2108	11.0944	22.36	102.2702
680	69.6723	464.7965	11.0348	14.56	97.9412
681	59.2194	342.4857	11.0944	22.36	100.8946
682	54.0773	356.8307	11.0944	22.36	100.5322
683	58.7699	353.5241	11.7567	15.86	104.2861
684	52.6582	415.0224	8.3049	19.34	121.3451
685	53.3334	486.1285	10.1350	32.54	97.9522

Alt.	Inter.	Cost	Vol.	Clk	Power
686	32.7925	367.8312	8.3522	33.76	143.5299
687	77.4386	306.1197	11.9253	22.36	103.7210
688	41.7973	324.8457	10.3008	32.66	119.4750
689	35.9020	420.1780	10.4039	22.36	114.3168
690	36.2471	398.5894	10.4039	22.36	114.5977
691	58.1086	358.3022	10.2150	27.32	112.0303
692	75.8932	335.2991	10.5993	34.24	98.1872
693	53.2840	453.9175	9.4822	12.64	112.6093
694	70.9123	332.5934	12.4531	14.22	103.5744
695	51.8407	486.1285	10.1350	35.34	97.1644
696	45.4728	334.1571	8.4218	14.76	152.9749
697	47.1477	309.6835	8.4218	14.76	150.3525
698	52.7044	445.1489	10.8741	34.74	97.9980
699	49.1599	309.6835	8.4218	14.76	150.9090
700	52.1134	323.3969	10.8741	16.36	113.7555
701	51.9008	314.0386	11.9046	27.46	120.2831
702	85.8078	439.7772	8.7246	16.36	110.4735
703	66.0124	393.3299	10.5993	34.24	95.9482
704	32.3721	367.8312	8.3522	33.76	146.1087
705	33.6619	367.8312	8.3522	33.76	142.2339
706	35.7984	376.2014	11.8506	40.14	110.0000
707	50.3310	419.6632	9.3450	34.74	98.2370
708	41.7323	437.9529	8.8679	21.64	111.6443
709	85.7364	428.1332	8.7660	34.74	101.9236
710	37.7287	418.9921	9.5167	25.84	103.9451
711	60.5479	359.5972	12.8873	24.24	99.2901
712	70.8090	335.2991	10.5993	34.24	98.6664
713	56.0900	337.5795	8.0212	26.04	137.0761
714	48.5282	344.4156	8.9205	35.76	113.5053
715	52.9530	486.1285	10.1350	34.74	98.1474
716	32.4091	350.5468	8.8063	33.76	144.0609
717	39.7721	354.0287	9.6860	28.12	121.1489
718	24.6758	418.9921	9.5167	17.84	118.8535
719	48.3957	411.6577	8.9965	27.82	110.0683
720	83.6300	420.7198	8.3104	16.36	111.6128
721	29.4069	501.2802	11.9789	16.36	129.9390
722	61.2228	351.9278	8.3724	16.36	114.3702
723	74.9191	329.8847	8.3104	16.36	112.9969
724	64.3295	402.8895	8.3049	31.74	108.0416
725	55.6362	384.2171	8.3724	21.84	126.6065
726	40.9411	373.5624	7.3652	37.34	121.9530
727	37.9495	344.7457	8.3522	33.76	143.6866
728	76.7510	307.7527	6.6126	21.64	126.0752
729	77.4139	450.0894	10.5993	31.74	92.9922
730	60.4863	436.4192	10.5993	34.24	96.4027
731	74.6355	489.6547	10.5993	30.86	92.9184
732	66.8877	489.6547	10.5993	30.86	93.6951
733	67.0392	543.1948	10.4877	30.86	95.1206
734	51.1322	301.4903	8.9205	35.76	113.5255
735	35.8955	309.0332	7.1941	29.36	148.4184

Alt.	Inter.	Cost	Vol.	Clk	Power
736	43.7715	363.4755	7.1941	29.36	146.3506
737	44.3436	486.1285	10.1350	34.74	98.8140
738	44.6899	486.1285	10.1350	34.74	98.4966
739	45.2492	486.1285	10.1350	34.74	98.1500
740	56.6510	337.5795	8.0212	26.04	136.8358
741	56.6398	435.5950	8.8679	23.94	112.7566
742	53.1392	436.1534	11.5592	34.74	97.2584
743	50.0217	419.6632	9.3450	34.74	100.6118
744	50.9795	419.6632	9.3450	34.74	99.7213
745	51.8675	419.6632	9.3450	34.74	100.2869
746	58.1391	427.1377	8.8679	23.94	112.6928
747	51.8962	419.6632	9.3450	34.74	98.8928
748	31.5042	324.9844	7.5432	28.96	157.2401
749	34.7229	353.8896	7.5432	28.96	154.5057
750	33.4837	324.9844	7.5432	28.96	155.1905
751	57.9870	337.5795	8.0212	26.04	136.8342
752	44.5500	417.1567	8.8679	25.44	111.8646
753	54.4385	315.3638	9.5167	29.64	106.5380
754	51.1948	328.8576	8.9205	35.76	111.7311
755	54.1360	437.9529	8.8679	21.64	109.2335
756	42.5396	398.7839	9.5167	29.64	105.1742
757	75.4714	307.7527	6.6126	21.64	127.7249
758	48.1243	349.5251	8.9205	35.76	114.3601
759	70.0117	332.5934	12.4531	14.22	104.7862
760	42.5880	342.4613	11.5696	29.64	108.0662
761	45.4115	352.0981	9.8255	35.76	115.7664
762	54.5222	437.9529	8.8679	21.64	110.4881
763	51.9789	437.9529	8.8679	20.34	110.8057
764	53.5922	410.6913	8.8679	23.94	118.8363
765	55.9434	410.6913	8.8679	23.94	113.1585
766	49.1422	419.6632	9.3450	34.74	101.7164
767	61.4118	351.3106	8.3174	26.94	129.1796
768	84.7519	303.1227	12.5620	15.86	123.2110
769	55.4554	489.6216	10.1350	34.74	97.0459
770	70.7453	381.5379	10.8658	26.76	97.2266
771	47.3348	430.7933	9.3450	34.74	98.8574
772	50.7573	346.3281	11.0944	22.36	101.4406
773	75.8788	347.6755	9.9838	22.14	107.5685
774	59.1558	390.0994	10.8977	31.76	96.3031
775	53.1513	486.1285	10.1350	34.74	98.2082
776	54.6040	394.0360	8.0280	29.64	119.8316
777	54.1810	408.4362	8.0280	28.54	112.2151
778	32.4362	400.3657	9.1355	31.96	118.0121
779	36.4853	393.5883	11.0456	15.86	124.0893
780	71.7176	332.5934	12.4531	14.22	102.8013
781	52.4143	407.8024	8.3287	13.56	143.8493
782	56.0778	337.5795	8.0212	26.04	139.0428
783	54.4882	355.2317	8.0212	26.04	138.3236
784	63.3621	312.3561	8.0438	22.44	120.4032
785	71.4266	321.4214	12.4531	14.72	102.3408

Alt.	Inter.	Cost	Vol.	Clk	Power
786	69.6072	358.5851	12.4531	14.22	102.8996
787	53.2987	490.4301	12.5707	14.24	98.8871
788	61.0634	380.2269	12.4531	28.24	98.6329
789	78.7545	288.1649	12.7738	19.46	123.1817
790	43.2931	327.5125	9.2938	14.76	151.2296
791	46.3333	461.6319	9.2299	11.34	115.8174
792	55.8510	286.2898	10.5547	34.42	125.6359
793	63.0196	312.3561	8.0438	22.44	122.8437
794	51.6559	325.4054	8.7134	36.56	119.8175
795	35.9105	309.0332	7.1941	29.36	148.7645
796	51.9938	419.6632	9.3450	34.74	98.8928
797	32.4151	338.8461	8.8063	33.76	147.5204
798	51.1447	315.3877	8.9205	35.76	113.3133
799	66.5361	351.1630	7.4892	23.02	129.0808
800	72.0615	332.5934	12.4531	14.22	103.0864
801	79.7382	288.1649	12.7738	19.46	123.7440
802	56.2363	356.8307	11.0944	22.36	101.0495
803	53.3041	486.1285	10.1350	34.74	98.0474
804	51.1877	328.8576	8.9205	35.76	112.1163

II Scalability case study - bandwidth input matrix

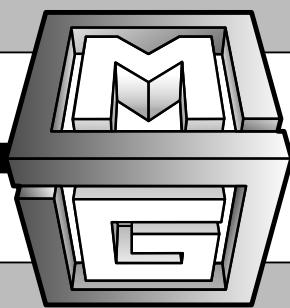
S	T	BW	T	BW	T	BW	T	BW
1	89	1	25	1690	49	19200	90	6.6
2	89	1	26	1690	50	19200	90	6.6
3	89	1	27	1690	51	19200	90	6.6
4	89	1	28	1690	52	19200	90	6.6
5	89	1	29	1690	53	19200	90	6.6
6	89	1	30	1690	54	19200	90	6.6
7	89	1	31	1690	55	19200	90	6.6
8	89	1	32	1690	56	19200	90	6.6
9	89	1	33	1690	57	19200	90	6.6
10	89	1	34	1690	58	19200	90	6.6
11	89	1	35	1690	59	19200	90	6.6
12	89	1	36	1690	60	19200	90	6.6
13	89	1	37	1690	61	19200	90	6.6
14	89	1	38	1690	62	19200	90	6.6
15	89	1.2	39	1690	63	19200	90	6.6
16	89	1.2	40	1690	64	19200	90	6.6
17	89	1.2	41	1690	65	19200	90	6.6
18	89	1.2	42	1690	66	19200	90	6.6
19	89	1.2	43	1690	67	19200	90	6.6
20	89	1.2	44	1690	68	19200	90	6.6
21	89	1.2	45	1690	69	19200	90	6.6
22	89	1.2	46	1690	70	19200	90	6.6
23	89	1.2	47	1690	71	19200	90	6.6
24	89	1.2	48	1690	72	19200	90	6.6
25	77	6.6	0	0	0	0	0	0
26	77	6.6	0	0	0	0	0	0
27	77	6.6	0	0	0	0	0	0
28	77	6.6	0	0	0	0	0	0
29	77	6.6	0	0	0	0	0	0
30	77	6.6	0	0	0	0	0	0
31	78	6.6	0	0	0	0	0	0
32	78	6.6	0	0	0	0	0	0
33	78	6.6	0	0	0	0	0	0
34	78	6.6	0	0	0	0	0	0
35	78	6.6	0	0	0	0	0	0
36	78	6.6	0	0	0	0	0	0
37	79	6.6	0	0	0	0	0	0
38	79	6.6	0	0	0	0	0	0
39	79	6.6	0	0	0	0	0	0
40	79	6.6	0	0	0	0	0	0
41	79	6.6	0	0	0	0	0	0
42	79	6.6	0	0	0	0	0	0
43	80	6.6	0	0	0	0	0	0
44	80	6.6	0	0	0	0	0	0
45	80	6.6	0	0	0	0	0	0

S	T	BW	T	BW	T	BW	T	BW
46	80	6.6	0	0	0	0	0	0
47	80	6.6	0	0	0	0	0	0
48	80	6.6	0	0	0	0	0	0
49	81	300	0	0	0	0	0	0
50	81	300	0	0	0	0	0	0
51	81	300	0	0	0	0	0	0
52	82	300	0	0	0	0	0	0
53	82	300	0	0	0	0	0	0
54	82	300	0	0	0	0	0	0
55	83	300	0	0	0	0	0	0
56	83	300	0	0	0	0	0	0
57	83	300	0	0	0	0	0	0
58	84	300	0	0	0	0	0	0
59	84	300	0	0	0	0	0	0
60	84	300	0	0	0	0	0	0
61	85	300	0	0	0	0	0	0
62	85	300	0	0	0	0	0	0
63	85	300	0	0	0	0	0	0
64	86	300	0	0	0	0	0	0
65	86	300	0	0	0	0	0	0
66	86	300	0	0	0	0	0	0
67	87	300	0	0	0	0	0	0
68	87	300	0	0	0	0	0	0
69	87	300	0	0	0	0	0	0
70	88	300	0	0	0	0	0	0
71	88	300	0	0	0	0	0	0
72	88	300	0	0	0	0	0	0
73	25	26.4	26	26.4	27	26.4	28	26.4
74	31	26.4	32	26.4	33	26.4	34	26.4
75	37	26.4	38	26.4	39	26.4	40	26.4
76	43	26.4	44	26.4	45	26.4	46	26.4
77	73	52.7	0	0	0	0	0	0
78	74	52.7	0	0	0	0	0	0
79	75	52.7	0	0	0	0	0	0
80	76	52.7	0	0	0	0	0	0
89	90	39.6	0	0	0	0	0	0

III. PROMETHEE-compatible presentations of multicriteria evaluation tables

III PROMETHEE-compatible presentations of multicriteria evaluation tables

See following article.



Université Libre de Bruxelles

CoDE - SMG

**PROMETHEE-compatible presentations of
multicriteria evaluation tables**

CoDE-SMG – Technical Report Series

Karim LIDOUH, N. Anh Vu DOAN, Yves DE SMET

CoDE-SMG – Technical Report Series

Technical Report No.
TR/SMG/2014-002

May 2014

CoDE-SMG – Technical Report Series
ISSN 2030-6296

Published by:

CoDE-SMG, CP 210/01
UNIVERSITÉ LIBRE DE BRUXELLES
Bvd du Triomphe
1050 Ixelles, Belgium

Technical report number TR/SMG/2014-002

The information provided is the sole responsibility of the authors and does not necessarily reflect the opinion of the members of CoDE-SMG. The authors take full responsibility for any copyright breaches that may result from publication of this paper in the CoDE-SMG – Technical Report Series. CoDE-SMG is not responsible for any use that might be made of data appearing in this publication.

PROMETHEE-compatible presentations of multicriteria
evaluation tables

CoDE-SMG – Technical Report Series

Karim LIDOUH

karim.lidouh@ulb.ac.be

N. Anh Vu DOAN

nguyen.anh.vu.doan@ulb.ac.be

Yves DE SMET

yves.de.smet@ulb.ac.be

CoDE-SMG, Université Libre de Bruxelles, Brussels, Belgium

May 2014

PROMETHEE-compatible presentations of multicriteria evaluation tables

**Karim Lidouh*,
N. Anh Vu Doan, and
Yves De Smet**

Department of Computer and Decision Engineering
École polytechnique de Bruxelles, Université libre de Bruxelles,
50 Avenue F.D. Roosevelt CP 210/01, 1050 Brussels, Belgium
E-mail: klidouh@ulb.ac.be
E-mail: ndoan1@ulb.ac.be
E-mail: yvdesmet@ulb.ac.be

*Corresponding author

Abstract: Most decision problems involve the simultaneous optimisation of several conflicting criteria. Generally, the first step to solve such problems is to identify the set of alternatives and the criteria they will be evaluated on, leading to the construction of an evaluation table. Of course, there are numerous ways to build such a table. For a problem of n alternatives and m criteria, there are $n! \cdot m!$ possibilities of representation. However, from a multicriteria point of view some of them can be more interesting than the others. In this article, we will focus on the PROMETHEE and GAIA methods from which the extracted information will serve to build tables. In order to evaluate the properties of these PROMETHEE-based representations, an indicator will be defined that uses only ordinal information of the values contained in a given table. This measure will also serve as a fitness function for a genetic algorithm that will find good – if not the best – tables. These will allow to draw comparisons with PROMETHEE-based representations.

Keywords: multicriteria decision aid, PROMETHEE, GAIA, evaluation table, visualisation, genetic algorithm

Reference to this paper should be made as follows: Lidouh, K., Doan, N.A.V and De Smet, Y. (xxxx) 'PROMETHEE-compatible presentations of multicriteria evaluation tables', *Int. J. Multicriteria Decision Making*, Vol. x, No. x, pp.XXX–XXX.

Biographical notes: Karim Lidouh has a degree as a Civil Engineer in Computer Science, a Master in Management and completed in 2014 a PhD thesis on the integration of multicriteria tools in geographical information systems. He works as a Teaching Assistant in the fields of statistics and quantitative methods at the Solvay Brussels School of Economics and Management (SBS-EM) of the Université libre de Bruxelles. He also gives occasional courses on operations research and optimisation methods at the IESEG School of Management.

Anh Vu Doan is a Teaching Assistant in the fields of statistics, computer science and decision engineering at the Engineering Faculty of the Université libre de Bruxelles. He obtained a master in Electrical Civil Engineering, with Electronics options in 2009 and started a PhD thesis on the application of multi-objective optimization and multicriteria decision tools to the design of

3D-stacked integrated circuits in microelectronics.

Yves De Smet is Assistant Professor at the Engineering Faculty of the Université libre de Bruxelles. He is both head of the Computer and Decision Engineering laboratory and of the SMG unit. Yves De Smet holds a degree in Mathematics (1998) and a PhD in Applied Sciences (2005). His research interests are focused on multicriteria decision aid and multi-objective optimization. Besides his academic activities he has been involved in different industrial projects. Since 2010, he has been co-founder of the Decision Sights spin-off.

1 Introduction

Most strategic decision problems involve the simultaneous optimisation of several conflicting criteria. For instance, in a procurement conducted by a transport company, the buyer (looking for new trucks) wants to simultaneously optimize: the investment and operational costs, both the quality of the vehicle and the supplier, the time of delivery, the mean time before failure, etc.

In a multicriteria analysis, the first step is to identify the set of alternatives, denoted $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ and evaluation criteria, denoted $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$. This leads to the construction of an evaluation table (see Table 1).

For the last 50 years several works have been proposed for the visual exploration of data tables or matrices. The first works dealt with reorderable matrices as a tool to represent structures and relationships [1, 2]. Later, other approaches such as block clustering were considered [3, 4] before the use of colour matrix visualisation [5, 6]. Throughout the years these techniques have been used to highlight trends and interesting displays in several cases such as the famous traveling salesman and shortest path problems [7, 8]. They have also been used conjointly with other visualisation tools such as scatterplot matrices and parallel coordinates [9]. In all of these contributions, the authors all agree on the fact that reordering rows and columns in data tables is an essential part in the graphical exploration of quantitative or qualitative data [10–12].

Table 1 Evaluation table

a	$f_1(\cdot)$	$f_2(\cdot)$	\dots	$f_j(\cdot)$	\dots	$f_m(\cdot)$
a_1	$f_1(a_1)$	$f_2(a_1)$	\dots	$f_j(a_1)$	\dots	$f_m(a_1)$
a_2	$f_1(a_2)$	$f_2(a_2)$	\dots	$f_j(a_2)$	\dots	$f_m(a_2)$
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
a_i	$f_1(a_i)$	$f_2(a_i)$	\dots	$f_j(a_i)$	\dots	$f_m(a_i)$
\vdots	\vdots	\vdots	\ddots	\vdots	\ddots	\vdots
a_n	$f_1(a_n)$	$f_2(a_n)$	\dots	$f_j(a_n)$	\dots	$f_m(a_n)$

From a multicriteria decision aid viewpoint, there are plenty of ways to represent evaluation tables. For instance, one may list the set of alternatives and criteria in an alphabetic order. Potentially, given a set of n alternatives and m criteria, $n! \cdot m!$ different evaluation tables can be built. For instance, for a limited problem with only 10 alternatives and 6 criteria, already $2.6127 \cdot 10^9$ different evaluation tables can be displayed. From a multicriteria point of view, some of them are more interesting than others. The aim of this paper is to investigate how to represent the evaluation tables in order to display as much multicriteria information as possible.

Since the late sixties, researchers working in Multicriteria Decision Aid (MCDA) have developed original methods to address these situations. For instance, we can mention Multi-Attribute Utility Theory (MAUT) [13], Analytical Hierarchy Process (AHP) [14], ELECTRE [15], PROMETHEE [16], MACBETH [17], etc.

In this contribution, we will focus ourselves on PROMETHEE methods. These have been applied in hundreds of applications in finance, health care, environmental management, transport, sports, hydrology and water management, production, etc. [18]. This success is due on their simplicity and the existence of user-friendly software.

By making use of information extracted using the PROMETHEE methodology, we will be able to build evaluation tables to convey additional characteristics of the problem. In most cases, these representations will focus on gathering similar alternatives and rearranging the criteria such that their strong and weak characteristics appear more clearly. Furthermore, out of the many possibilities that stem from using the PROMETHEE methodology, we will identify those that yield the most relevant results. We will do so using a subset of the ranking of best cities [19]. This subset, composed of 14 cities is given in table 2. Each of these cities has been evaluated using six criteria, the details of which are described in a later section where we make use of the full ranking. Even with a small set like this one, we have $14! \cdot 6!$ (i.e. $6.2768 \cdot 10^{13}$) possible representations of this evaluation table.

Table 2 Best cities ranking subset - Evaluation table

Perm		1	2	3	4	5	6
	City	Stability	Healthcare	Culture and Environment	Education	Infrastructure	Spatial Characteristics
1	Hong Kong	95	87.5	85.9	100	96.4	75
2	Stockholm	95	95.8	91.2	100	96.4	58.9
3	Rome	80	87.5	91.7	100	92.9	67.3
4	New York	70	91.7	91.7	100	89.3	65.2
5	Atlanta	85	91.7	91.7	100	92.9	42.9
6	Buenos Aires	70	87.5	85.9	100	85.7	42.3
7	Santiago	75	70.8	89.1	83.3	85.7	35.1
8	Sao Paulo	60	70.8	80.3	66.7	66.1	52.4
9	Mexico City	45	66.7	82.4	75	46.4	65.8
10	New Delhi	55	58.3	55.6	75	58.9	58.6
11	Istanbul	55	50	68.8	58.3	67.9	47.5
12	Jakarta	50	45.8	59.3	66.7	57.1	42.3
13	Tehran	50	62.5	35.9	50	33.9	53.6
14	Dakar	50	41.7	59.7	50	37.5	22.6

Source: [19].

Of course, one can imagine that these representations are not all interesting. Therefore, we need to evaluate the tables in order to choose the best representation(s). For that

purpose, we have defined an indicator that only uses the ordinal properties of the values contained in a table: the ∇ -indicator. This will be described in Section 3.1.

With this measure, it will be possible to find the best permutations on the alternatives and the criteria. However, since the number of possibilities can be huge even with a small dataset, it may be impossible to find the best representation in reasonable time. Therefore we have decided to use a genetic algorithm (GA) for the optimisation of the ∇ -indicator, this family of algorithms having shown good properties for similar situations [20]. GAs belong to the class of evolutionary algorithms which generate solutions to optimisation problems using techniques inspired by natural evolution. Details about the implementation will be given in Section 3.2.

We will apply these two approaches on two case studies: the best cities ranking by the Economist Intelligence Unit and the Environmental Performance Index by two research centres of the Columbia University.

This paper is organized as follows: in Section 2 we will give a brief description of the PROMETHEE and GAIA methodologies and identify the possible evaluation tables that can be derived from them. Next, in Section 3, we will define the ∇ -indicator that will allow to evaluate the different representations. This measure will also be used as a fitness function for the genetic algorithm that will be applied. Finally, in Section 4 we will illustrate the two approaches using the previously-described case studies.

2 Constructivist approach

2.1 PROMETHEE and GAIA

In this subsection we recall the basics of the PROMETHEE and GAIA methods. Of course, a detailed description of these approaches goes beyond the scope of this contribution. Therefore we refer the interested reader to [21] for a detailed analysis.

Let $\mathcal{A} = \{a_1, a_2, \dots, a_n\}$ be a set of n alternatives and $\mathcal{F} = \{f_1, f_2, \dots, f_m\}$ be a set of m criteria. Without loss of generality, we assume that all criteria have to be maximized. The PROMETHEE methods are based on pairwize comparisons. At first, each pair of alternatives $a_i, a_j \in \mathcal{A}$ is compared on every criterion f_k :

$$d_k(a_i, a_j) = f_k(a_i) - f_k(a_j)$$

The quantity $d_k(a_i, a_j)$ represents the *advantage* of a_i over a_j for criterion f_k . On the one hand, when $d_k(a_i, a_j)$ is small enough, there is no good reason to say that a_i is better than a_j regarding criterion f_k . On the other hand, when $d_k(a_i, a_j)$ exceeds a certain limit, the decision maker may express that a_i is strictly preferred to a_j for f_k . In order to model these statements, the difference $d_k(a_i, a_j)$ is transformed into a unicriterion preference degree, denoted $P_k(a_i, a_j)$, by using a non-decreasing function H_k :

$$P_k(a_i, a_j) = H_k(d_k(a_i, a_j)), \quad \forall a_i, a_j \in \mathcal{A}$$

The quantity $P_k(a_i, a_j) \in [0, 1]$ and $P_k(a_i, a_j) = 0$ when $d_k(a_i, a_j) < 0$. There are plenty of functions that can be considered to compute the unicriterion preference degrees. In most software implementing the PROMETHEE method, 6 main functions are considered [22]. Figure 1 represents the so-called linear preference function. Two thresholds characterize it:

- q_k plays the role of an *indifference* threshold. When the difference $d_k(a_i, a_j) \leq q_k$, it is considered to be so small that the unicriterion preference is equal to zero;
- p_k plays the role of a *preference* threshold. When the difference $d_k(a_i, a_j) \geq p_k$, it is considered to be important enough to state that a_i is strongly preferred to a_j for this criterion.

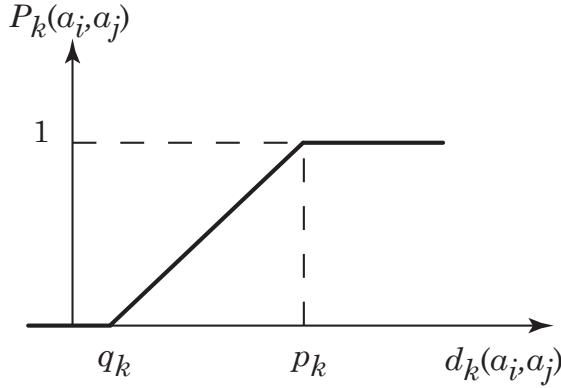


Figure 1: Generalized criterion of type 5

Once the unicriterion preference degrees between two actions a_i and a_j have been computed for every criterion, one has to aggregate these marginal contributions to obtain $P(a_i, a_j)$ i.e. a global measure of the preference of a_i over a_j :

$$P(a_i, a_j) = \sum_{k=1}^m \omega_k \cdot P_k(a_i, a_j)$$

where ω_k represents the relative importance of criterion f_k . These weights are assumed to be positive and normalized. Obviously, we have $P(a_i, a_j) \geq 0$ and $P(a_i, a_j) + P(a_j, a_i) \leq 1$.

The PROMETHEE I and II rankings are based on the exploitation of the matrix P . Therefore, three flows are built.; the positive flow ϕ^+ , the negative flow ϕ^- and the net flow ϕ :

$$\phi^+(a_i) = \frac{1}{n-1} \sum_{a_j \in \mathcal{A}, i \neq j} P(a_i, a_j)$$

$$\phi^-(a_i) = \frac{1}{n-1} \sum_{a_j \in \mathcal{A}, i \neq j} P(a_j, a_i)$$

$$\phi(a_i) = \phi^+(a_i) - \phi^-(a_i)$$

The PROMETHEE I ranking is obtained as the intersection of the rankings induced by ϕ^+ and ϕ^- . The PROMETHEE II ranking is given by the ranking given by ϕ .

Finally, it is worth noting that:

$$\phi(a_i) = \frac{1}{n-1} \sum_{k=1}^m \sum_{a_j \in A} [P_k(a_i, a_j) - P_k(a_j, a_i)] \cdot \omega_k = \sum_{k=1}^m \phi_k(a_i) \cdot \omega_k$$

where $\phi_k(a_i)$ is called the k^{th} unicriterion net flow assigned to action a_i .

The PROMETHEE I and II ranking provide prescriptive tools for decision making. The GAIA [23] tool complements them with a descriptive approach. The idea is to represent each alternative by its evaluations in the unicriterion net flow space:

$$\Phi(a_i) = [\phi_1(a_i), \phi_2(a_i), \dots, \phi_m(a_i)]$$

GAIA is the result of a principal component analysis applied on this dataset. Therefore, the decision maker is able to visualize the decision problem on a plane and compare:

- the relative positions of alternatives (in order to identify groups of similar or distinct alternatives profiles);
- the relative positions of criteria (in order to identify conflicts or redundancies);
- the relative positions of alternatives with respect to a given criterion (in order to identify the best and worst alternatives for the different points of views);
- the relative positions of alternatives with respect to the so-called *decision stick* (in order to identify the best compromise solutions).

2.2 Visualisation possibilities

To illustrate the different combinations of orders we could use to rearrange an evaluation table, let us consider the subset of cities we introduced in Section 1. Using this table of 14 alternatives and 6 criteria, we will apply the PROMETHEE methodology by setting some arbitrary values for the parameters that the method requires. To keep a ranking close to the one obtained by the Economist Intelligence Unit, we will use the same weight values as they did for their model (see Section 4.2). Then, for the sake of simplicity, we will make use of usual preference functions. These are generalized preference functions for which both thresholds are equal to 0. By doing so, we will only make use of the ordinal data extracted from our table. Of course, a detailed discussion on the parameters goes beyond the scope of this contribution. The following example is just used for illustration purposes. Figure 2 shows the GAIA plane we obtain for this case.

Ordering the alternatives

One obvious choice to order the alternatives would be to use the PROMETHEE II net flows we obtain. This would allow us to order the alternatives from best to worst thereby ensuring that the best profiles are at the top of the table while the ones with the less desirable ones are at the end. Grouping the alternatives with similar global scores would inevitably serve to gather alternatives with similar profiles at the top and bottom of the table. In problems where the few best and few worst alternatives are of interest to the user, this representation could give interesting insights.

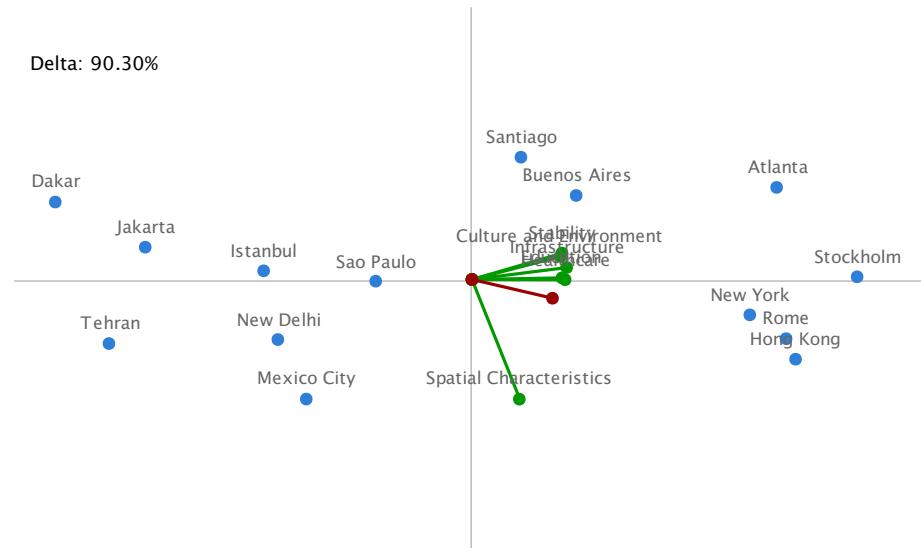


Figure 2: GAIA plane for the best cities subset

Two other options to order the alternatives can be found by using the GAIA plane. Indeed, in this projection, the alternatives have been positioned such that similar alternatives are closer to each other. By using the angle on which each alternative is positioned and scanning the entire plane we would order the alternatives by selecting them based on the types of strong points (or weak points) they have. The information that will be highlighted in this table are the profiles of the alternatives.

One other use of the GAIA plane would be to select alternatives based on their accordance with the decision axis. To do so easily we can compute the angles between the decision axis and the alternatives and select the alternatives from the smallest to the greatest angle. This would generate an order that is similar to the PROMETHEE II ranking obtained using the net flows. When considering the GAIA plane we obtained in Figure 2, we can see that the first alternatives would be Hong Kong, Rome, and New York. The last ones encountered would be Istanbul, Jakarta, and Dakar.

Ordering the criteria

To order the criteria, we also have different options. One would be to order them based on their weights. This would make sure that the first criteria are the ones that hold the greatest importance. This however will not necessarily order the criteria in a way that gathers similar characteristics of the profiles. Therefore, unless the profiles are already similar due to the nature of the alternatives and their order, the obtained table would not be so easy to read.

Other orders involve the use of the GAIA plane once more. The first one consists in choosing the criteria in the order they appear when scanning the projections. Since the criteria are positioned based on their correlation, choosing this order would allow us

to group strong and weak points that usually appear simultaneously. This type of order would work best if coupled with the similar way of ordering the alternatives (i.e. based on the angle of their position). Let us note that there are several options for the use of this technique. Indeed the starting angle has to be chosen but also the direction by which we will scan the plane.

Another possibility would be to select criteria based on their proximity with the decision axis. Once again, an easy way to do so, would be to compute the angle between the criteria axis and the decision axis and then to order them from smallest to greatest angle. The results using this technique however could be rather unpredictable.

Illustration

Among all the combinations of orders we generated only four drew our attention:

- Netflow - Angle: The first one consists in ordering the alternatives based on their net flow and the alternatives based on the angle of their axis on the GAIA plane. This representation has proven after several simulations to give us the most expected results by grouping all the good and bad alternatives and displaying their profiles such that the variations in their evaluations are smoother and easier to compare.
- Netflow - Weight: The second combination orders the alternatives based on their net flow and the criteria based on their weight. Even if this representation's aim is not to display smoother profiles like the previous one, it can be useful to attract the reader's attention on the characteristics that will have a greater impact on the final decision. Therefore it was only natural that the alternatives be ordered according to the final ranking we obtain.
- Angle - Angle: Among the combinations that use the angle at which the alternatives are located, the only one that gave us meaningful results were the ones where the criteria as well were ordered according to the angle of their axes. By choosing to start at the position of the decision axis and scanning the plane in an anti-clockwise motion, the alternatives appear from best to worst to best based on the characteristics of their profiles.
- Proximity - Proximity: Finally ordering the criteria and alternatives based on their proximity to the decision axis can sometimes give us interesting representations. However in most cases the results do not reflect any particular relationship between the alternatives or the criteria aside from the approximate ranking from best to worst.

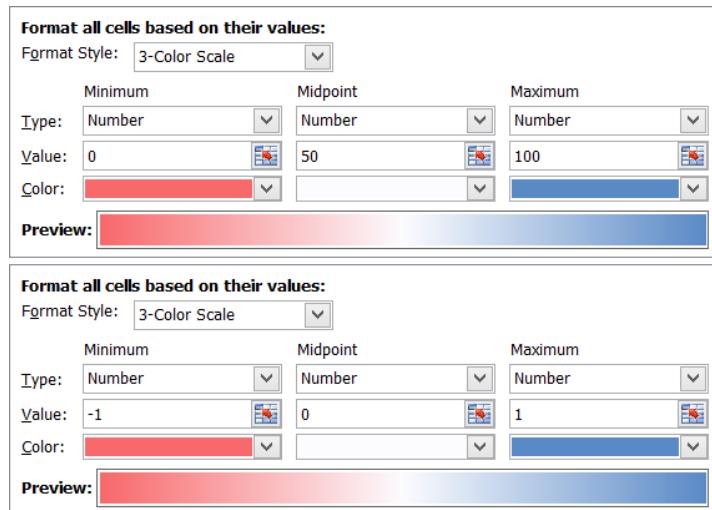
Table 3 shows the possible combinations of orders that can be applied to our evaluation tables where \bigcirc indicates the four chosen combinations and \times indicates the other possibilities that have not been kept for the study.

To better compare these tables and understand the impact of changing the orders we applied a colouring process to the value cells based on two colourmaps. A spreadsheet software such as Microsoft Excel allows us to use conditional formatting rules to achieve this result. The colourmaps we used are displayed in Figure 3. The lowest values will be coloured in red, the middle values in white and the highest values in blue. The evaluations in our examples range from 0 to 100. For the net flows, these values range from -1 to +1. Furthermore, we did not use pure red and blue colours for the extreme values as these would have rendered the data unreadable. Instead we chose the threshold colours

Table 3 Combinations of orders and chosen representations

		Order of criteria		
		Weights	Angle	Proximity
Order of alternatives	Netflows	○	○	×
	Angle	×	○	×
	Proximity	×	×	○

commonly proposed by Excel for readability purposes: the red we use is [248, 105, 107] and the blue is [90, 138, 198] in the RGB colour system.

**Figure 3:** Colourmaps for the conditional formatting rules

Thus, for the best cities ranking subset, we have generated the four chosen combinations. These are given in Figures 4, 5, and 6.

The first table (see Figure 4) shows the table for alternatives ordered by the net flow and criteria ordered by their angle relative to the decision stick. As can be seen due to the coloring, the choice of starting at the position of the decision stick seems wrong in this case. Indeed criterion 2 would have been better off as the first in this table.

The second table (see Figure 5) which uses the weights to order the criteria gives surprisingly good results. Indeed, as the criteria with the greatest weight are also the ones with the smallest evaluation values, it seems as though the best values are located in the top right corner and the lowest in the bottom left.

Unsurprisingly, the table in Figure 6a shows the best values in the four corners and the worst in the middle of the table. This is the natural result when ordering the alternatives and the criteria based on the angle of their position on the GAIA plane.

	Crit4	Crit5	Crit3	Crit1	Crit6	Crit2	NetFlows
A2	100	96,4	91,2	95	58,9	95,8	0,704808
A1	100	96,4	85,9	95	75	87,5	0,694231
A3	100	92,9	91,7	80	67,3	87,5	0,667308
A4	100	89,3	91,7	70	65,2	91,7	0,541346
A5	100	92,9	91,7	85	42,9	91,7	0,446154
A6	100	85,7	85,9	70	42,3	87,5	0,030769
A7	83,3	85,7	89,1	75	35,1	70,8	-0,03846
A8	66,7	66,1	80,3	60	52,4	70,8	-0,14615
A9	75	46,4	82,4	45	65,8	66,7	-0,17885
A10	75	58,9	55,6	55	58,6	58,3	-0,30865
A11	58,3	67,9	68,8	55	47,5	50	-0,35481
A13	50	33,9	35,9	50	53,6	62,5	-0,575
A12	66,7	57,1	59,3	50	42,3	45,8	-0,65577
A14	50	37,5	59,7	50	22,6	41,7	-0,82692

Figure 4: Visualisation - Best cities ranking subset - Netflow-angle

	Crit6	Crit1	Crit3	Crit2	Crit5	Crit4	NetFlows
A2	58,9	95	91,2	95,8	96,4	100	0,704808
A1	75	95	85,9	87,5	96,4	100	0,694231
A3	67,3	80	91,7	87,5	92,9	100	0,667308
A4	65,2	70	91,7	91,7	89,3	100	0,541346
A5	42,9	85	91,7	91,7	92,9	100	0,446154
A6	42,3	70	85,9	87,5	85,7	100	0,030769
A7	35,1	75	89,1	70,8	85,7	83,3	-0,03846
A8	52,4	60	80,3	70,8	66,1	66,7	-0,14615
A9	65,8	45	82,4	66,7	46,4	75	-0,17885
A10	58,6	55	55,6	58,3	58,9	75	-0,30865
A11	47,5	55	68,8	50	67,9	58,3	-0,35481
A13	53,6	50	35,9	62,5	33,9	50	-0,575
A12	42,3	50	59,3	45,8	57,1	66,7	-0,65577
A14	22,6	50	59,7	41,7	37,5	50	-0,82692

Figure 5: Visualisation - Best cities ranking subset - Netflow-Weight

	Crit4	Crit5	Crit3	Crit1	Crit6	Crit2	NetFlows
A2	100	96,4	91,2	95	58,9	95,8	0,704808
A5	100	92,9	91,7	85	42,9	91,7	0,446154
A6	100	85,7	85,9	70	42,3	87,5	0,030769
A7	83,3	85,7	89,1	75	35,1	70,8	-0,03846
A14	50	37,5	59,7	50	22,6	41,7	-0,82692
A12	66,7	57,1	59,3	50	42,3	45,8	-0,65577
A11	58,3	67,9	68,8	55	47,5	50	-0,35481
A8	66,7	66,1	80,3	60	52,4	70,8	-0,14615
A13	50	33,9	35,9	50	53,6	62,5	-0,575
A10	75	58,9	55,6	55	58,6	66,7	-0,30865
A9	75	46,4	82,4	45	65,8	66,7	-0,17885
A1	100	96,4	85,9	95	75	87,5	0,694231
A3	100	92,9	91,7	80	67,3	87,5	0,667308
A4	100	89,3	91,7	70	65,2	91,7	0,541346

(a) Angle-Angle

(b) Proximity-Proximity

Figure 6: Visualisation - Best cities ranking subset

As for Figure 6b, it shows the table that is obtained when both the alternatives and the criteria are ordered based on their proximity to the decision axis. The result shows us that the highest values are gathered in the top left corner. Let us note that a very similar result would have been obtained had we used the net flows to order the alternatives.

3 Optimisation approach

3.1 Development of an optimisation indicator: the ∇ -indicator

In order to compare the different possibilities of visualisation, we need an indicator that will evaluate the presentation quality of a table. We arbitrarily chose to verify that the best values are located at the top left of the table while the worst values are located at the bottom right. We have therefore developed such an indicator that will only use the ordinal information of the values contained in the table. We will denote such indicator ∇ that is the total number of ordered pairs of values where the first value is greater than or equal to the second, for each row and column (in other words, the number of pairs that are compatible with our convention).

A computation example of the ∇ -indicator is shown in Table 4 where:

- $\nabla_{i \cdot}$ is the ∇ value of the i -th row:

$$\nabla_{i \cdot} = \sum_{k=1}^m \sum_{l=k+1}^m \mathbb{1}_{k < l} \mathbb{1}_{f_k(a_i) \geq f_l(a_i)}$$

- $\nabla_{\cdot j}$ is the ∇ value of the j -th column:

$$\nabla_{\cdot j} = \sum_{k=1}^n \sum_{l=k+1}^n \mathbb{1}_{k < l} \mathbb{1}_{f_j(a_k) \geq f_j(a_l)}$$

- The total $\nabla = \sum_{i=1}^n \nabla_{i \cdot} + \sum_{j=1}^m \nabla_{\cdot j}$

We can use this indicator as a value to compare the different representations of an evaluation table.

3.2 Optimizing the ∇ -indicator with a genetic algorithm

Now that we have defined an indicator that can evaluate the presentation quality of an evaluation table, we can use it in order to find the best permutations of alternatives and criteria that will maximize it. As stated in the introduction, finding the optimal solution might be tedious as the number of solutions can be huge: $n! \cdot m!$ possibilities.

Finding the best solution table can thus be seen as a combinatorial optimisation problem. Up to now, we have not found an exact method to solve it, therefore we will use a genetic algorithm to find a good solution in reasonable time. In what follows we describe its main steps.

Table 4 Best cities ranking subset - Evaluation table - ∇ -indicator computation

City	Stability	Healthcare	Culture and Environment	Education	Infrastructure	Spatial Characteristics	∇_i
Hong Kong	95	87.5	85.9	100	96.4	75	9
Stockholm	95	95.8	91.2	100	96.4	58.9	8
Rome	80	87.5	91.7	100	92.9	67.3	6
New York	70	91.7	91.7	100	89.3	65.2	9
Atlanta	85	91.7	91.7	100	92.9	42.9	7
Buenos Aires	70	87.5	85.9	100	85.7	42.3	9
Santiago	75	70.8	89.1	83.3	85.7	35.1	8
Sao Paulo	60	70.8	80.3	66.7	66.1	52.4	10
Mexico City	45	66.7	82.4	75	46.4	65.8	7
New Delhi	55	58.3	55.6	75	58.9	58.6	4
Istanbul	55	50	68.8	58.3	67.9	47.5	8
Jakarta	50	45.8	59.3	66.7	57.1	42.3	8
Tehran	50	62.5	35.9	50	33.9	53.6	9
Dakar	50	41.7	59.7	50	37.5	22.6	12
$\nabla_{\cdot j}$	82	83	76	88	84	66	$\nabla = 593$

Source: [19].

Selection

A solution table is composed of two informations: the permutation on the alternatives and the permutation on the criteria. These will thus constitute a gene. For example, the gene $\{[7, 1, 13, 2, 14, 6, 10, 12, 11, 4, 8, 3, 9, 5], [3, 5, 6, 1, 4, 2]\}$ applied to the example of Table 2 will produce the Table 5.

Table 5 Best cities ranking subset - Evaluation table - Permutation example

Perm	City	3	5	6	1	4	2
		Culture and Environment	Infrastructure	Spatial Characteristics	Stability	Education	Healthcare
7	Santiago	75	70.8	89.1	83.3	85.7	35.1
1	Hong Kong	95	87.5	85.9	100	96.4	75
13	Tehran	50	62.5	35.9	50	33.9	53.6
2	Stockholm	95	95.8	91.2	100	96.4	58.9
14	Dakar	50	41.7	59.7	50	37.5	22.6
6	Buenos Aires	70	87.5	85.9	100	85.7	42.3
10	New Delhi	55	58.3	55.6	75	58.9	58.6
12	Jakarta	50	45.8	59.3	66.7	57.1	42.3
11	Istanbul	55	50	68.8	58.3	67.9	47.5
4	New York	70	91.7	91.7	100	89.3	65.2
8	Sao Paulo	60	70.8	80.3	66.7	66.1	52.4
3	Rome	80	87.5	91.7	100	92.9	67.3
9	Mexico City	45	66.7	82.4	75	46.4	65.8
5	Atlanta	85	91.7	91.7	100	92.9	42.9

From this point, a pool of random solutions can be generated to initialize the algorithm (by using a uniform distribution). From a set of 10000 randomly-generated solutions, the 100 best will be selected and will constitute the initial population. This selection rule will be used for each generation.

Crossover

For the crossover, a classical method has been used: the one-point crossover. A random crossover point is selected on both parents. Beyond that point, the data will be completed following the order of appearance in the other parent, as shown in Figure 7. This method will be used separately both for the alternatives and criteria permutations. The crossover probability has been set as $p_c = 1$.

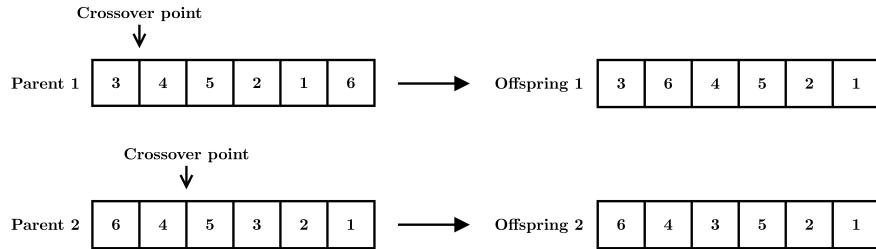


Figure 7: Crossover example

Mutation

For the mutation, two random data of an offspring will be swapped, as shown in Figure 8. This method will be used separately both for the alternatives and criteria permutations. The mutation probability has been set as $p_m = 0.1$.

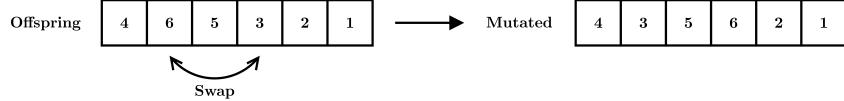


Figure 8: Mutation example

Termination conditions

Common termination conditions have been used: fixed number of generations (50) and maximum computation time (30 min).

Simulation environment and performance

The genetic algorithm have been implemented on MATLAB and the simulations have been carried out on an Intel® Core™ i5-2410M Processor (dual core, 2.3 GHz).

The average running time for 50 generations, as a function of the problem size (using our case studies as examples) is given in Table 6. Figure 9 shows the evolution of the ∇ -indicator over 50 iterations for the best cities ranking subset. The other case studies have similar convergence shape.

Table 6 Genetic algorithm - Average running time

	Table size (n alt. \times m crit.)		
	14×6	70×6	19×9
Time (min.)	5.75	25.01	10.56

Figure 9: Evolution of the ∇ -indicator over the iterations for the best cities ranking subset

4 Case studies

4.1 Comparison of the approaches

4.1.1 Comparison of the tables: defining a ratio

When comparing different representations, we can compare their ∇ -indicators. However, depending on the size of a table, this comparison is not that trivial. Indeed, a difference of, for instance, 20 between two ∇ -indicators can be important for a small table but insignificant for a big table.

Therefore, in order to keep the comparison as objective as possible, we introduce a ratio denoted R :

$$R = \frac{\nabla - \nabla_{\text{worst}}}{\nabla_{\text{best}} - \nabla_{\text{worst}}}$$

where ∇_{best} is the best ∇ found with our genetic algorithm and ∇_{worst} is the value associated to the worst table found by taking the opposite permutations of the alternatives and criteria of the best table.

Let us note that it is possible to find a theoretical maximum value for the ∇ -indicator of a given table for n alternatives and m criteria:

$$\nabla_{\text{max}} = \frac{n \cdot m \cdot (m - 1)}{2} + \frac{m \cdot n \cdot (n - 1)}{2}$$

However it would not be realistic to use it as a reference. Indeed this ∇_{max} can only be reached with well-chosen values which would not be the case for real multicriteria decision problems.

4.2 Best cities ranking

Our first case study is based on the best cities ranking by the Economist Intelligence Unit [19]. This dataset is composed of 70 cities evaluated on 6 criteria: stability, healthcare, culture and environment, education, infrastructure, and spatial characteristics (see Table 7).

This study consists in an update of the existing EIU Liveability index to which a sixth criterion has been added to take into account spatial characteristics. This added factor carries a weight of 25% and seeks to account for spatial aspects such as urban form, the geographical situation of the city, cultural assets and pollution.

Just like we illustrated our approach in Section 2.2, we make use of usual preference functions. The weights for the criteria are taken from the analysis by the EIU (see Table 7).

Table 7 Best cities ranking - Evaluation table

Perm	City	1	2	3	4	5	6	Aggregate
		Stability (18.75%)	Healthcare (15%)	Culture and Environment (18.75%)	Education (7.5%)	Infrastructure (15%)	Spatial Characteristics (25%)	
1	Hong Kong	95	87.5	85.9	100	96.4	75	87.8
2	Amsterdam	80	100	97.2	91.7	96.4	71.3	87.4
3	Osaka	90	100	93.5	100	96.4	64	87.4
4	Paris	85	100	97.2	100	96.4	63.7	87.1
5	Sydney	90	100	94.4	100	100	55.7	86
6	Stockholm	95	95.8	91.2	100	96.4	58.9	86
7	Berlin	85	100	97.2	91.7	96.4	61.7	85.9
8	Toronto	100	100	97.2	100	89.3	50	85.4
9	Munich	85	100	97.2	91.7	89.3	62.5	85.1
10	Tokyo	90	100	94.4	100	92.9	53.3	84.3
11	Rome	80	87.5	91.7	100	92.9	67.3	83.6
12	London	70	87.5	97.2	100	89.3	72.6	83.5
13	Madrid	85	87.5	94.4	100	92.9	61.3	83.5
14	Washington DC	80	91.7	94.4	100	96.4	55.1	82.2
15	Chicago	85	91.7	91.7	100	92.9	52.7	81.5
16	New York	70	91.7	91.7	100	89.3	65.2	81.3
17	Los Angeles	80	91.7	94.4	100	89.3	50.3	79.9
18	San Francisco	85	91.7	94.4	83.3	85.7	53	79.7
19	Boston	80	91.7	91.7	100	96.4	46.7	79.6
20	Seoul	80	83.3	85.6	100	89.3	58.8	79.1
21	Atlanta	85	91.7	91.7	100	92.9	42.9	79
22	Singapore	95	87.5	76.6	83.3	100	46.7	78.2
23	Miami	85	91.7	91.7	100	92.9	39.3	78.1
24	Budapest	85	91.7	90	100	83.9	43	77.4
25	Lisbon	80	87.5	95.1	91.7	80.4	41.7	75.3
26	Buenos Aires	70	87.5	85.9	100	85.7	42.3	73.3
27	Moscow	65	79.2	81.5	91.7	83.9	54.2	72.3
28	St Petersburg	65	87.5	81.5	83.3	80.4	48.2	70.9
29	Athens	75	83.3	83.1	75	75	47.3	70.8
30	Beijing	80	66.7	72.2	83.3	85.7	51.5	70.5
31	Santiago	75	70.8	89.1	83.3	85.7	35.1	69.3
32	Warsaw	80	70.8	80.3	75	82.1	39	68.4
33	Shanghai	80	62.5	75	75	75	46.1	66.8
34	Shenzhen	85	62.5	63.7	66.7	82.1	48.5	66.7
35	Lima	60	66.7	81.7	91.7	75	47.3	66.5
36	Sao Paulo	60	70.8	80.3	66.7	66.1	52.4	64.9
37	Kuala Lumpur	80	62.5	67.8	91.7	76.8	36.6	64.6
38	Tianjin	90	66.7	65.3	66.7	82.1	27.7	63.4
39	Guangzhou	80	62.5	61.1	66.7	76.8	42.9	63.1
40	Johannesburg	50	58.3	90.5	83.3	69.6	44.9	63
41	Mexico City	45	66.7	82.4	75	46.4	65.8	62.9
42	Rio de Janeiro	55	66.7	77.5	83.3	71.4	44	62.8
43	Bucharest	80	66.7	74.3	66.7	66.1	34.7	62.5
44	Kiev	70	75	73.4	83.3	50	33.3	60.2
45	Belgrade	60	75	73.1	75	57.1	36	59.4
46	New Delhi	55	58.3	55.6	75	58.9	58.6	58.6
47	Dalian	85	62.5	62	66.7	75	21	58.4
48	Manila	60	58.3	63.2	66.7	64.3	46.1	58
49	Bangkok	50	62.5	64.4	100	69.6	36.3	57.8
50	Bogota	35	62.5	75.2	66.7	64.3	50.6	57.3
51	Istanbul	55	50	68.8	58.3	67.9	47.5	57.1
52	Mumbai	60	54.2	56.3	66.7	51.8	52.1	55.7
53	Casablanca	65	45.8	60.9	58.3	60.7	43.8	54.9
54	Caracas	30	41.7	76.6	75	60.7	52.1	54
55	Cairo	55	45.8	54.9	58.3	53.6	48.2	51.9
56	Jakarta	50	45.8	59.3	66.7	57.1	42.3	51.5
57	Hanoi	55	54.2	53.7	58.3	51.8	38.4	50.2
58	Tashkent	50	58.3	55.3	75	51.8	26.8	48.6
59	Damascus	55	50	54.2	41.7	55.4	36.5	48.5
60	Ho Chi Minh City	55	50	49.5	66.7	48.2	35.1	48.1
61	Tehran	50	62.5	35.9	50	33.9	53.6	47.7
62	Nairobi	40	45.8	69.9	66.7	42.9	33.9	47.4
63	Lusaka	60	33.3	59.7	41.7	55.4	23.2	44.7
64	Phnom Penh	60	37.5	49.3	58.3	53.6	24.1	44.6
65	Karachi	20	45.8	38.7	66.7	51.8	48.5	42.8
66	Dakar	50	41.7	59.7	50	37.5	22.6	41.9
67	Abidjan	25	45.8	54.2	50	53.6	30.1	41
68	Dhaka	50	29.2	43.3	41.7	26.8	35.7	37.9
69	Lagos	25	33.3	52.3	33.3	48.2	22.3	34.8
70	Harare	30	20.8	53	66.7	35.7	17.3	33.4

Source: [19].

The GAIA plane obtained using this parametrisation is shown in Figure 10. We can observe that all the five initial criteria of this ranking seem to be correlated whereas the newly-added category discriminates the cities in a different way.

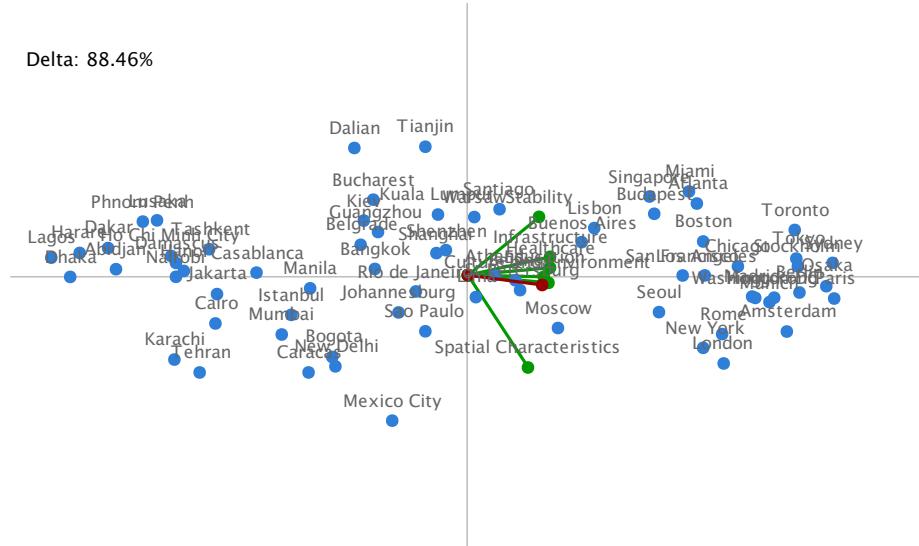


Figure 10: GAIA plane for the best cities ranking

Two PROMETHEE-based tables are given in Figures 11 and 12, respectively for the evaluations and the unicriterion netflows, alongside the two best tables found with our genetic algorithm.

We can see in Figure 11a a representation of the evaluation table using the Netflow-Angle combination as described in Section 2.2. This illustration is similar to the one obtained for the best cities subset (see Figure 4). Once again, since the starting point for the scanning process has been arbitrarily set as the decision axis, criterion 6 (spatial characteristics) ends up in the middle of the table at the fourth position. The computed ∇ -indicator for this representation is 13149 which is unsurprisingly lower than the one for the best table (13535, see Table 11b) due to the incurred penalty of the sixth criterion's position. However, this representation still holds good ordinal properties: if we evaluate the ratio R of this table, as defined in Section 4.1.1, the value found is 96.35%. That means that ∇ -indicator is at 96.35% of the possible range for this evaluation table.

The Figure 12a illustrates the Angle-Angle combination for the unicriterion netflow table of this case. The same observations as for Figure 6a applies: by ordering both alternatives and criteria based on their angle relatively to the decision axis, the lowest unicriterion netflow values will lie in the center of the table while the highest will be located at the top and bottom. That is why the ∇ -indicator (7722) is low compared to the best found (13289). As expected, the ratio R is close to 50% (44.83%).

G20, excluding the European Union member. We have thus an evaluation table (see Table 8) composed of 19 countries evaluated on 9 criteria: air quality, health impacts, water sanitation, agriculture, biodiversity habitat, climate energy, fisheries, forests, water resources.

Table 8 Environmental Performance Index (G20) - Evaluation table

Perm	Country	1 Air Quality	2 Health Impacts	3 Water Sanitation	4 Agriculture	5 Habitat	6 Energy	7 Climate	8 Fisheries	9 Forests	Water Resources	EPI Ranking
1	Australia	98.33	100	100	66.46	83.08	47.67	19.37	100	92.33	3	
2	Germany	78.5	100	100	65.31	100	62.77	13.4	31.35	95.18	6	
3	United Kingdom	95.82	100	100	66.03	70.11	54.24	0	43.06	97.93	12	
4	Italy	80.85	100	63.51	58.87	79.77	63.41	24.93	55.41	91.44	22	
5	Canada	97.85	100	95.9	62.52	58.4	59.85	21.54	16.64	80.42	24	
6	Japan	84.79	99.2	100	46.48	73.53	43.54	25.34	55.41	71.26	26	
7	France	89.44	100	100	65.55	54.45	49.83	0	37.94	83.8	27	
8	USA	96.41	95.33	86.48	61.53	63.35	56.45	3.34	14.35	63.66	33	
9	Saudi Arabia	84.45	94.68	83.48	92	93.7	46.63	6.43	0	28.54	35	
10	South Korea	62.24	96.93	85.92	46.98	50.4	41.55	22.24	33.76	83.68	43	
11	Mexico	87.09	76.67	46.2	55.21	62.32	51.35	25.34	19.87	37.45	65	
12	Turkey	84.07	66.06	71.43	56.67	32.62	46.52	21.9	52.35	48.93	66	
13	South Africa	94.4	47.51	36.08	79.2	63.96	49.87	2.52	100	27.86	72	
14	Russia	94.36	83.12	45.17	16.93	53.39	61.02	12.73	35.07	21.5	73	
15	Brazil	97.64	68.59	50.44	74.51	66.74	53.82	24.68	10.81	10.87	77	
16	Argentina	99.64	85.07	75.7	96	44.88	16.79	15.68	0	11.75	93	
17	Indonesia	75.31	67.55	24.29	51.85	78.08	45.25	25.8	7.75	0.02	112	
18	China	18.81	76.23	33.15	33.85	66.63	65.16	14.68	25.34	18.18	118	
19	India	23.24	50.04	26.28	58.4	39.18	35.24	22.64	35.07	10.49	155	

Source: [24].

The GAIA plane is represented in Figure 13. We can see that the criteria are separated in three groups. Two of them are clearly opposed: fisheries vs. air quality and agriculture. For instance, Indonesia, China and India have some of the best evaluations regarding fisheries while their scores on air quality and agriculture are amongst the worst of our set. The opposite features can be observed for countries such as UK, Canada, France and the USA.



Figure 13: GAIA plane for the Environmental Performance Index (G20)

For this case, we will analyse the two other combinations: the Netflow-Weight combination for the evaluation table and the Proximity-Proximity combination for the unicriterion netflow table, alongside the two best tables found with our genetic algorithm, as shown respectively in Figures 14 and 15.

The first combination, in Figure 14a does not display any particular structure in the organisation of the evaluations. Indeed, in this case, we have enriched the table by displaying the criteria with the highest weight first. The ∇ -indicator reaches a high value (1556) compared to the best found (1619, Figure 14b). However this is a coincidence given that the weights are not linked to the evaluations.

In the second combination (Figure 15a), the alternatives and criteria are ordered depending on their proximity to the decision axis. We can therefore observe that the best countries are located on the top of the table which contributes to the high value of the obtained ∇ -indicator (1417).

The best found table (Figure 15b) with a ∇ -indicator of 1465 does not seem to hold better ordinal properties. This is explained by the heterogeneous nature of the profiles, as observed in the GAIA plane. When comparing this table with the best one for the evaluations, this heterogeneity is emphasized. A particular example of that statement can be observed for criterion 7 (fisheries) where the evaluations range only between 0 and 25.8 whereas the corresponding unicriterion netflows will range between -0.944 and 1.

	Crit1	Crit2	Crit3	Crit5	Crit6	Crit9	Crit7	Crit8	Crit4	NetFlows
A1	98,33	100	100	83,08	47,67	92,33	19,37	100	66,46	0,625926
A3	95,82	100	100	70,11	54,24	97,93	0	43,06	66,03	0,518519
A2	78,5	100	100	100	62,77	95,18	13,4	31,35	65,31	0,474074
A5	97,85	100	95,9	58,4	59,85	80,42	21,54	16,64	62,52	0,37963
A4	80,85	100	63,51	79,77	63,41	91,44	24,93	55,41	58,87	0,353703
A7	89,44	100	100	54,45	49,83	83,8	0	37,94	65,55	0,246297
A6	84,79	99,2	100	73,53	43,54	71,26	25,34	55,41	46,48	0,225
A8	96,41	95,33	86,48	63,35	56,45	63,66	3,34	14,35	61,53	0,161111
A9	84,45	94,68	83,48	93,7	46,63	28,54	6,43	0	92	-0,02963
A15	97,64	68,59	50,44	66,74	53,82	10,87	24,68	10,81	74,51	-0,06759
A11	87,09	76,67	46,2	62,32	51,35	37,45	25,34	19,87	55,21	-0,14352
A14	94,36	83,12	45,17	53,39	61,02	21,5	12,73	35,07	16,93	-0,17315
A10	62,24	96,93	85,92	50,4	41,55	83,68	22,24	33,76	46,98	-0,18056
A16	99,64	85,07	75,7	44,88	16,79	11,75	15,68	0	96	-0,18518
A13	94,4	47,51	36,08	63,96	49,87	27,86	2,52	100	79,2	-0,23611
A18	18,81	76,23	33,15	66,63	65,16	18,18	14,68	25,34	33,85	-0,34537
A12	84,07	66,06	71,43	32,62	46,52	48,93	21,9	52,35	56,67	-0,3537
A17	75,31	67,55	24,29	78,08	45,25	0,02	25,8	7,75	51,85	-0,51667
A19	23,24	50,04	26,28	39,18	35,24	10,49	22,64	35,07	58,4	-0,75278

(a) Netflow-Weight ($\nabla = 1556$)

	Crit2	Crit1	Crit5	Crit3	Crit9	Crit4	Crit6	Crit8	Crit7	Netflows
A1	100	98,33	83,08	100	92,33	66,46	47,67	100	19,37	0,625926
A3	100	95,82	70,11	100	97,93	66,03	54,24	43,06	0	0,518519
A2	100	78,5	100	100	95,18	65,31	62,77	31,35	13,4	0,474074
A4	100	80,85	79,77	63,51	91,44	58,87	63,41	55,41	24,93	0,353703
A5	100	97,85	58,4	95,9	80,42	62,52	59,85	16,64	21,54	0,37963
A7	100	89,44	54,45	100	83,8	65,55	49,83	37,94	0	0,246297
A6	99,2	84,79	73,53	100	71,26	46,48	43,54	55,41	25,34	0,225
A8	95,33	96,41	63,35	86,48	63,66	61,53	56,45	14,35	3,34	0,161111
A9	94,68	84,45	93,7	83,48	28,54	92	46,63	0	6,43	-0,02963
A15	68,59	97,64	66,74	50,44	10,87	74,51	53,82	10,81	24,68	-0,06759
A11	76,67	87,09	62,32	46,2	37,45	55,21	51,35	19,87	25,34	-0,14352
A10	96,93	62,24	50,4	85,92	83,68	46,98	41,55	33,76	22,24	-0,18056
A16	85,07	99,64	44,88	75,7	11,75	96	16,79	0	15,68	-0,18518
A13	47,51	94,4	63,96	36,08	27,86	79,2	49,87	100	2,52	-0,23611
A12	66,06	84,07	32,62	71,43	48,93	56,67	46,52	52,35	21,9	-0,3537
A14	83,12	94,36	53,39	45,17	21,5	16,93	61,02	35,07	12,73	-0,17315
A18	76,23	18,81	66,63	33,15	18,18	33,85	65,16	25,34	14,68	-0,34537
A17	67,55	75,31	78,08	24,29	0,02	51,85	45,25	7,75	25,8	-0,51667
A19	50,04	23,24	39,18	26,28	10,49	58,4	35,24	35,07	22,64	-0,75278

(b) Best found table ($\nabla = 1619$)**Figure 14:** Environmental Performance Index - Evaluations

5 Conclusion

Adequately reordered tables can give us interesting insights into the problems we tackle. This contribution shows how to achieve such results based on the PROMETHEE methodology. Several possibilities of tables have been proposed, taking into account information such as the netflows, the weight or the position of the alternatives and criteria on the GAIA plane.

Furthermore, an indicator has been proposed to evaluate the ordinal properties of these tables and compare them. It also serves as a fitness function for a genetic algorithm that

	Crit3	Crit2	Crit9	Crit5	Crit6	Crit8	Crit1	Crit4	Crit7	NetFlows
A3	0,777778	0,722222	1	0,333333	0,333333	0,444444	0,444444	0,444444	-0,944444	0,518519
A1	0,777778	0,722222	0,777778	0,777778	-0,22222	0,944444	0,888889	0,555556	0	0,625926
A2	0,777778	0,722222	0,888889	1	0,777778	-0,11111	-0,555556	0,222222	-0,333333	0,474074
A7	0,777778	0,722222	0,555556	-0,44444	-0,11111	0,333333	0,111111	0,333333	-0,944444	0,246297
A5	0,444444	0,722222	0,333333	-0,33333	0,555556	-0,444444	0,777778	0,111111	0,111111	0,37963
A8	0,333333	0,111111	0,111111	-0,11111	0,444444	-0,555556	0,555556	0	-0,666667	0,161111
A4	-0,22222	0,722222	0,666667	0,666667	0,888889	0,722222	-0,44444	-0,11111	0,666667	0,353703
A6	0,777778	0,333333	0,222222	0,444444	-0,666667	0,722222	-0,11111	-0,777778	0,833333	0,225
A9	0,111111	0	-0,22222	0,888889	-0,33333	-0,944444	-0,22222	0,888889	-0,555556	-0,02963
A10	0,222222	0,222222	0,444444	-0,666667	-0,777778	0	-0,777778	-0,666667	0,333333	-0,18056
A16	0	-0,11111	-0,666667	-0,777778	-1	-0,944444	1	1	-0,11111	-0,18518
A13	-0,666667	-1	-0,333333	0	0,944444	0,333333	0,777778	-0,777778	-0,23611	
A18	-0,777778	-0,444444	-0,555556	0,111111	1	-0,22222	-1	-0,888889	-0,22222	-0,34537
A15	-0,33333	-0,555556	-0,777778	0,222222	0,222222	-0,666667	0,666667	0,666667	0,555556	-0,06759
A14	-0,555556	-0,22222	-0,44444	-0,555556	0,666667	0,166667	0,222222	-1	-0,44444	-0,17315
A11	-0,44444	-0,33333	-0,11111	-0,22222	0,111111	-0,33333	0	-0,44444	0,833333	-0,14352
A12	-0,11111	-0,777778	0	-1	-0,444444	0,555556	-0,333333	-0,333333	0,222222	-0,3537
A17	-1	-0,666667	-1	0,555556	-0,555556	-0,777778	-0,666667	-0,555556	1	-0,51667
A19	-0,88889	-0,88889	-0,88889	-0,88889	-0,88889	0,166667	-0,88889	-0,22222	0,444444	-0,75278

(a) Proximity-Proximity ($\nabla = 1417$)

	Crit3	Crit2	Crit8	Crit9	Crit4	Crit1	Crit6	Crit7	Crit5	Netflows
A1	0,777778	0,722222	0,944444	0,777778	0,555556	0,888889	-0,22222	0	0,777778	0,625926
A3	0,777778	0,722222	0,444444	1	0,444444	0,444444	0,333333	-0,944444	0,333333	0,518519
A2	0,777778	0,722222	-0,11111	0,888889	0,222222	-0,555556	0,777778	-0,33333	1	0,474074
A4	-0,22222	0,722222	0,722222	0,666667	-0,11111	-0,44444	0,888889	0,666667	0,666667	0,353703
A5	0,444444	0,722222	-0,44444	0,333333	0,111111	0,777778	0,555556	0,111111	-0,33333	0,37963
A7	0,777778	0,722222	0,333333	0,555556	0,333333	0,111111	-0,11111	-0,944444	-0,44444	0,246297
A6	0,777778	0,333333	0,722222	0,222222	-0,777778	-0,11111	-0,666667	0,833333	0,444444	0,225
A8	0,333333	0,111111	-0,555556	0,111111	0	0,555556	0,444444	-0,666667	-0,11111	0,161111
A9	0,111111	0	-0,94444	-0,22222	0,888889	-0,22222	-0,33333	-0,555556	0,888889	-0,02963
A15	-0,33333	-0,555556	-0,666667	-0,777778	0,666667	0,666667	0,222222	0,555556	0,222222	-0,06759
A11	-0,44444	-0,33333	-0,33333	-0,11111	-0,44444	0	0,111111	0,833333	-0,22222	-0,14352
A13	-0,666667	-1	0,944444	-0,33333	0,777778	0,333333	0	-0,777778	0	-0,23611
A10	0,222222	0,222222	0	0,444444	-0,666667	-0,777778	-0,777778	0,333333	-0,666667	-0,18056
A16	0	-0,11111	-0,94444	-0,666667	1	1	-1	-0,11111	-0,777778	-0,18518
A12	-0,11111	-0,777778	0,555556	0	-0,33333	-0,33333	-0,44444	0,222222	-1	-0,3537
A14	-0,555556	-0,22222	0,166667	-0,44444	-1	0,222222	0,666667	-0,44444	-0,555556	-0,17315
A18	-0,777778	-0,44444	-0,22222	-0,555556	-0,88889	-1	1	-0,22222	0,111111	-0,34537
A17	-1	-0,666667	-0,777778	-1	-0,555556	-0,666667	-0,555556	1	0,555556	-0,51667
A19	-0,88889	-0,88889	0,166667	-0,88889	-0,22222	-0,88889	-0,88889	0,444444	-0,88889	-0,75278

(b) Best found table ($\nabla = 1465$)**Figure 15:** Environmental Performance Index - Unicriterion netflows

will search for better representations. When comparing the obtained tables, we can note that the results with the ∇ -indicators and the ratio R seem consistent.

Indeed, as shown in Figure 16, the combination Netflow-Angle has a ratio R of approximately 95% in most cases. This is mostly due to the size of the used dataset. For the best cities ranking, the high number of alternatives will have a greater impact on the ∇ -indicator whereas in small sets, such as the EPI, the position of the criteria can have more influence, especially when they are conflicting with the decision axis. The same statement applies for the Netflow-Weight and Proximity-Proximity combinations.

	Best cities ranking subset				Best cities ranking				Environmental Performance Index			
	Evaluations		Unicriterion Flows		Evaluations		Unicriterion Flows		Evaluations		Unicriterion flows	
	Nabla	R	Nabla	R	Nabla	R	Nabla	R	Nabla	R	Nabla	R
GA Best	648	100,00%	603	100,00%	13535	100,00%	13289	100,00%	1619	100,00%	1465	100,00%
GA Worst	149	0,00%	205	0,00%	2962	0,00%	3199	0,00%	643	0,00%	838	0,00%
Netflow-Angle	625	95,39%	589	96,48%	13149	96,35%	13209	99,21%	1392	76,74%	1430	94,42%
Netflow-Weight	533	76,95%	597	98,49%	12959	94,55%	13186	98,98%	1556	93,55%	1439	95,85%
Angle-Angle	424	55,11%	388	45,98%	7662	44,45%	7722	44,83%	1176	54,61%	1214	59,97%
Proxi-Proxi	611	92,59%	567	90,95%	12695	92,06%	12475	91,93%	1503	88,11%	1417	92,34%

Figure 16: Comparison of the obtained results based on the ∇ -indicator and the ratio R

For the Angle-Angle combination, it is interesting to note the values are all close to 50%. This is in accordance with the way these tables are built. Scanning the GAIA plane and going from the best to the worst to the best alternatives ensures that only half of the table will respect the ordinal properties that we defined for the ∇ -indicator. All the tables used for these computations are available in the appendices (see Sections A, B, and C).

To the best of our knowledge, this is the first attempt at reordering tables using multicriteria information. This study has shown that multicriteria-enriched tables can still hold interesting ordinal properties. This could be easily generalised to other methodologies. Instead of netflows, other aggregators could be considered.

References and Notes

- 1 J Bertin. *Semiology of Graphics: Diagrams, Networks, Maps (Translation of "Sémiologie Graphique. Les Diagrammes, les Réseaux, les Cartes (1967)" by William J. Berg)*. Madison, Wisconsin. University of Wisconsin Press, 1967.
- 2 John W Carmichael and Peter HA Sneath. Taxometric maps. *Systematic Biology*, 18(4):402–415, 1969.
- 3 John A Hartigan. Direct clustering of a data matrix. *Journal of the american statistical association*, 67(337):123–129, 1972.
- 4 Robert Tibshirani, Trevor Hastie, Mike Eisen, Doug Ross, David Botstein, Pat Brown, et al. Clustering methods for the analysis of dna microarray data. *Dept. Statist., Stanford Univ., Stanford, CA, Tech. Rep*, 1999.
- 5 Edward J Wegman. Hyperdimensional data analysis using parallel coordinates. *Journal of the American Statistical Association*, 85(411):664–675, 1990.
- 6 Michael Minnotte and W West. The data image: a tool for exploring high dimensional data sets. In *Proceedings of the ASA Section on Statistical Graphics*, pages 25–33. Citeseer, 1998.
- 7 JK Lenstra. Clustering a data array and the traveling-salesman problem. *Operations Research*, 22(2):413–414, 1974.
- 8 JR Slagle, CL Chang, and SR Heller. A clustering and data-reorganizing algorithm. *Systems, Man and Cybernetics, IEEE Transactions on*, (1):125–128, 1975.
- 9 Catherine B Hurley. Clustering visualizations of multidimensional data. *Journal of Computational and Graphical Statistics*, 13(4), 2004.
- 10 Chun-Houh Chen. Generalized association plots: Information visualization via iteratively generated correlation matrices. *Statistica Sinica*, 12(1):7–30, 2002.

- 11 Michael Friendly and Ernest Kwan. Effect ordering for data displays. *Computational Statistics & Data Analysis*, 43(4):509–539, 2003.
- 12 Han-Ming Wu, ShengLi Tzeng, and Chun-houh Chen. Matrix visualization. In *Handbook of Data Visualization*, Springer Handbooks Comp.Statistics, pages 681–708. Springer Berlin Heidelberg, 2008.
- 13 J.S. Dyer. MAUT – multiattribute utility theory. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 265–285. Springer Verlag, Boston, Dordrecht, London, 2005.
- 14 T.L. Saaty. The Analytic Hierarchy and Analytic Network Processes for the Measurement of Intangible Criteria and for Decision-Making. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 345–408. Springer Verlag, Boston, Dordrecht, London, 2005.
- 15 J. Figueira, V. Mousseau, and B. Roy. ELECTRE methods. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 133–162. Springer Verlag, Boston, Dordrecht, London, 2005.
- 16 J.P Brans and Ph. Vincke. A preference ranking organization method. *Management Science*, 31(6):647–656, 1985.
- 17 C.A. Bana e Costa, J.M. De Corte, and J.C. Vansnick. On the mathematical foundation of MACBETH. In J. Figueira, S. Greco, and M. Ehrgott, editors, *Multiple Criteria Decision Analysis: State of the Art Surveys*, pages 409–443. Springer Verlag, Boston, Dordrecht, London, 2005.
- 18 Majid Behzadian, R.B. Kazemzadeh, A. Albadvi, and M. Aghdasi. PROMETHEE: A comprehensive literature review on methodologies and applications. *European Journal of Operational Research*, 200(1):198–215, 2010.
- 19 Economist Intelligence Unit. Best cities ranking and report: A special report from the economist intelligence unit, 2012. http://pages.eiu.com/rs/eiu2/images/EIU_BestCities.pdf.
- 20 El-Ghazali Talbi. *Metaheuristics: From Design to Implementation*. Wiley Publishing, 2009.
- 21 J. Figueira, S. Greco, and M. Ehrgott. *Multiple criteria decision analysis: state of the art surveys*, volume 78. Springer Verlag, 2005.
- 22 Quantin Hayez, Yves De Smet, and Jimmy Bonney. D-Sight: A New Decision Making Software to Address Multi-Criteria Problems. *IJDSSST*, 4(4):1–23, 2012.
- 23 Bertrand Mareschal and Jean-Pierre Brans. Geometrical representations for MCDA. *European Journal of Operational Research*, 34(1):69–77, February 1988.
- 24 A. Hsu, J. Emerson, M. Levy, A. de Sherbinin, L. Johnson, O. Malik, J. Schwartz, and M. Jaiteh. The 2014 environmental performance index, 2014. <http://www.epi.yale.edu/>.

Résumé

Ces dernières décennies, l'industrie en microélectronique s'est astreinte à suivre la loi de Moore pour améliorer la performance des circuits intégrés (*Integrated Circuit, IC*). Cependant, il sera sans doute impossible de suivre cette loi dans le futur à cause de limitations physiques apparaissant avec la miniaturisation des transistors en-dessous d'un certain seuil si aucune innovation n'a lieu. Afin de surmonter ce problème, de nouvelles technologies ont émergées, et parmi elles les circuits 3D (*3D-Stacked Integrated Circuit, 3D-SIC*) ont été proposés pour maintenir l'évolution de la loi de Moore. Les 3D-SIC peuvent apporter de nombreux avantages dans le design des futurs IC mais au coût d'une complexité de design accrue étant donné leur nature fortement combinatoire, et l'optimisation de plusieurs critères conflictuels. Dans cette thèse, nous présentons une première étude des outils qui pourraient aider dans le design de 3D-SIC, en utilisant l'optimisation multi-objectifs (*multiobjective optimization, MOO*) et l'aide multicritère à la décision (*multi-criteria decision aid, MCDA*). Notre étude vise l'une des problématiques principales dans le design de 3D-SIC : le partitionnement avec estimation du *floorplanning* en tenant compte de plusieurs objectifs. Cette thèse montre que l'utilisation d'un paradigme multicritère peut fournir une analyse pertinente et objective du problème. Cela peut permettre une exploration rapide de l'espace de design et une amélioration des flots de conception actuels étant donné qu'il est possible de fournir des informations qualitatives et quantitatives par rapport à l'espace de design qui ne seraient pas disponibles avec les outils actuels. De même, de par sa flexibilité, la MOO peut tenir compte des multiples degrés de liberté des 3D-SIC, ce qui permet plus de possibilités de design qui ne sont généralement pas prises en compte avec les outils actuels. De plus, les algorithmes développés peuvent montrer des propriétés de robustesse même si le problème est complexe. Enfin, appliquer l'aide multicritère à la décision pourrait permettre aux designers de faire des choix pertinents selon un processus transparent.