FACULTAD DE INGENIERIA
INSTITUTO DE MATEMÁTICA Y ESTADÍSTICA - IMERL
PEDECIBA Informática

UNIVERSIDAD DE LA REPÚBLICA

P H D T H E S I S

to obtain the title of

PhD on Informatics

of the Universidad de la República

Defended by
Luis Alberto STÁBILE SUÁREZ
(lstabile@fing.edu.uy)

GRASP/VND Optimization Algorithms for Hard Combinatorial Problems

Doctoral Advisor: Franco ROBLEDO

Thesis Advisor: Pablo ROMERO
Thesis Co-Advisor: Mathias BOUREL

Defended on October 7, 2019

Jury:

Reviewers :	Dr. Michal PIORO	-	Warsaw University of Technology (Poland) and Lund University (Sweden)
	Dr. Martín MATAMALA	-	University of Chile (Chile)
President :	Dr. Marcos VIERA	-	Universidad de la República (PEDECIBA INFORMÁTICA)
	Dr. José Rafael LEÓN	-	Universidad de la República (PEDECIBA MATEMÁTICA)
	Dr. Antonio MAUTTONE	-	Universidad de la República (PEDECIBA INFORMÁTICA)

FUNDAMENTACIÓN DE LA TESIS DE DOCTORADO EN INFORMÁTICA

DEL MSc. ING. LUIS ALBERTO STÁBILE SUÁREZ

"GRASP/VND OPTIMIZATION ALGORITHMS FOR HARD COMBINATORIAL PROBLEMS"

La tesis de Luis Stábile se inscribe en el área de Optimización, y sus aportes son esencialmente metodológicos. Se identifican 3 contribuciones principales:

- Una resolución eficiente para abordar un problema combinatorio de alta complejidad computacional, que relacionado al clásico problema del clique máximo. La heurística propuesta es competitiva con las mejores técnicas hasta la fecha, e incorpora cotas para reducir el tiempo de cómputo.
- Una resolución heurística basada en estructuras de vecindades para asistir en el diseño de redes de alta confiabilidad. Dentro de los resultados obtenidos se destacan nuevas redes cúbicas candidatas a ser uniformemente confiables, invitando a nuevas líneas de investigación.
- Un estudio matemáticamente riguroso de la separabilidad en sistemas binarios estocásticos, que incluyen resultados teóricos profundos, y una completa caracterización de grafos separables.

En la presentación oral, el candidato ha demostrado experiencia, respondiendo satisfactoriamente las preguntas formuladas.

Por lo tanto, este tribunal califica esta tesis como aprobada con mención.

Montevideo, 7 de Octubre de 2019

DR. MICHAL PIORO **DR. MARTÍN MATAMALA** **DR. MARCOS VIERA**

DR. JOSÉ RAFAEL LEÓN **DR. ANTONIO MAUTTONE**

Julio Herrera y Reissig 565 (Instituto de Computación, Facultad de Ingeniería, Universidad de la República)
Montevideo 11300, Uruguay, Casilla de Correo 16120, Distrito 6 - Montevideo - Uruguay
Teléfonos: (+598) 2714 27 14 (Interno 12109), http://www.pedeciba.edu.uy/informatica / pedeciba_inf@fing.edu.uy

WARSAW UNIVERSITY of TECHNOLOGY
Faculty of Electronics and Information Technology
INSTITUTE of TELECOMMUNICATIONS
Ul. Nowowiejska 15/19, 00-665 Warsaw, Poland

Prof. Michał Pióro
m.pioro@tele.pw.edu.pl
+48 695 440 018

Warsaw, 20.08.2019

Report on the PhD thesis of Luis Alberto Stábile Suárez entitled "GRASP/VND Optimization Algorithms for Hard Combinatorial Problems"

SUMMARY

Generally speaking, the presented PhD thesis is devoted to solving hard combinatorial problems related to graphs. For the first two of them (finding a max cut-clique and finding a uniformly most-reliable graph) heuristic algorithms based on the GRASP (greedy randomized adaptive search procedure) and VND (variable neighborhood descent) approaches, customized to the considered problems by means of problem-specific properties, are presented. The third problem, consisting in computing system reliability, is also hard and is approached by theoretical means that allow for distinguishing a class of systems, for which this problem is tractable.

CONTENT AND CONTRIBUTIONS

Chapter 1 presents a brief summary of the three main problems studied in the thesis and gives some motivation why there are worth considering. The links between the problems, however, are not discussed.

Chapters 2, 3, and 4 constitute the first part of the thesis (**Part I**), devoted to an extension of the classical NP-complete maximum clique problem. The extension, called the maximum cut-clique (MCC) problem, asks for finding a clique with the maximum number of links in the cut corresponding to the clique (i.e., the set of all links with only one end node in the clique) and is studied in Chapters 2 and 3. The MCC problem is shown to be NP-complete (an original contribution) and treated by a heuristic method called GRASP/VND developed by the PhD Candidate (together with his co-authors). The method is based on smartly defined neighborhood structures, based on a set of originally derived inequalities. The presented numerical results demonstrate that, for the considered network instances, the proposed method almost always gives an optimal solution (that is independently found by means of an appropriate exact mixed-integer programming formulation) in a short time. An attempt to find reasonable lower and upper bounds on the maximum cut-clique is presented. The results of Part I constitute a non-trivial and seemingly effective technique for finding the cliques in question.

NIP: 525-000-58-34, REGON: 000001554
Bank PeKaO S.A. IV o/Warsaw, Poland
nr: 81 1240 1053 1111 0000 0500 5664
e-mail: sekretariat@tele.pw.edu.pl
tel: +48 22 234 72 32
+48 22 234 78 68
fax: +48 22 234 74 77

A version of MCC, maximum edge-weight neighborhood clique (MEWNC), is dealt with in Chapter 4, where appropriate extensions of the approach developed for MCC are discussed. The presented numerical results show that the extended methods are basically of quality similar to those developed for MCC.

In **Part II**, consisting of **Chapters 5 and 6**, the so called uniformly most-reliable graphs are studied. Finding such graphs is an important and interesting problem in reliability theory. Chapter 5 contains a nice summary of (deep) graph-theoretical results underlying this problem, and, using some known and new properties (like Proposition 1) proposes a heuristic GRASP/VND approach to find the graphs in question, which seems to be effective for the special case of 3-regular graphs. Considerations of Part II represent a good theoretical level. Chapter 6 is an extension of Chapter 5. It introduces the so called Cubic algorithm that effectively solves the problem of transforming a (2*r*,3*r*) graph to a highly reliable cubic graph. The construction is based on an originally proven theorem.

Part III is composed of **Chapter 7**. The paper included in this chapter is mathematically rigorous and contains non-trivial results, valuable for reliability analysis. It introduces an interesting concept of separable stochastic binary systems and describes an original approach to represent and analyze such systems, and to approximate characteristics of general systems by means of separable systems. A special attention is devoted to reliability analysis of the stochastic binary systems relevant for the classical problem of all-terminal reliability evaluation.

Finally, **Part IV** briefly summarizes the work described in the thesis.

EVALUATION AND COMMENTS

The presented thesis is of good quality and contains valuable results, both theoretical and algorithmic, in the fields of combinatorial optimization and reliability theory. The mathematical presentation is rigorous, the presented results correct, and the numerical results convincing.

As far as drawbacks of the presented thesis are concerned, I have noticed the following:

- Reading Part I is a bit tedious as the three papers presented there contain a lot of repetitions.
- Concerning Parts I and II, the following question could be discussed: Are there any polynomial approximate algorithms, i.e., algorithms producing solutions with guaranteed quality in polynomial time? Did the authors encounter any network examples for which their heuristics do not work well? Has the “real-life product-placement” application mentioned as the future work in all three papers composing Part I been finally studied?
- In general, potential applications of the problems and models studied in the paper could be described in a more convincing way.

The above drawbacks have little influence on the overall technical value of the thesis. However, taking them into account would somewhat increase readability of the thesis and make it more complete.

RECOMMENDATION

In my opinion, the results of PhD Candidate’s research described in the thesis constitute a valuable contribution to the field of combinatorial optimization and reliability theory. This judgment is implied by the technical contents of the thesis, including six valuable publications co-authored by the Candidate: two papers in international journals and four papers in the proceedings of good international conferences.

Therefore, I recommend Luis Alberto Stábile Suárez to present his thesis at a public defense leading to the Doctor Degree of the Universidad de la República.

Michał Pióro

Full Professor, Computer Networks and Services Division

Warsaw, 20.08.2019

Report on Luis Stábile's Ph. D. Thesis: GRASP/VND optimization algorithms for hard combinatorial problems

Contents

The thesis is composed of three parts each of those focusing in a different problem, related by the methodology used to solve them.

The first part is devoted to the study of the Max Cut Clique (MCC) problem and its weigthed version (MEWNC). From the complexity point of view it is proved that both problems are NP-hard. From a combinatorial perspective, upper bounds for the value of optimal solutions are given in terms of an easy to compute function of the degrees. The same tool allows to get lower and upper bounds for the size of the smallest clique achieving the optimum. In order to solve instances of the problem, a metaheuristic is proposed based on GRASP. In the construction phase a clique is built which then is locally modified by using four local search rules. The performance of the proposed algorithm is compared with the solution obtained from an integer lineal program formulation solved with commercial solver CPLEX.

In the second part, the problem of finding reliable network under edges failure is considered. The work focuses on the computation of cubic graphs minimizing the number of their disconnected spanning subgraphs, aiming to get a uniformly most-reliable graph. This goal is achieved for small size graphs.

In the last part, a reasonable generalization of previous work to the setting of boolean functions is given. The special cases of monotone boolean functions and separable boolean functions are studied. The concept of ``separable graphs'' appears when these notions are applied to the case of all reliability problem, and a combinatorial characterization for them is given, proving that the membership problem of separable graphs belongs to the class P.

Beauchef 851, torre norte, piso 5, Santiago – Chile. Tel: 6711530 Email: mar.mat.vas@dim.uchile.cl

Evaluation

Overall the thesis is well written and contains original results ranging over the areas of complexity theory and graph theory, and mainly, the area of metaheuristics. The contribution in terms of publications is above the average, and it would be quite strong if the two submitted papers are accepted.
Therefore, without any doubt, I recommend that Luis Stabile defends his doctoral thesis.

Additional comments

It would be interesting to evaluate the integrality gap of the integer linear programs that appear in this thesis. It would also be important to know if the interaction between the surgery and cubic movements does not prevent the Local Search Phase to finish.

If possible the thesis should include in the introduction of each chapter the notation not defined in the individual section.

Martín Matamala.
Full Professor
Universidad de Chile

Contents

Abstract

Two hard combinatorial problems are addressed in this thesis. The first one is known as the "Max Cut-Clique", a combinatorial problem introduced by P. Martins in 2012. Given a simple graph, the goal is to find a clique $\mathcal{C}$ such that the number of links shared between $\mathcal{C}$ and its complement $\mathcal{C}^{\mathcal{C}}$ is maximum.

In a first contribution, a GRASP/VND methodology is proposed to tackle the problem. In a second one, the $\mathcal{NP}$-Completeness of the problem is mathematically proved. Finally, a further generalization with weighted links is formally presented with a mathematical programming formulation, and the previous GRASP is adapted to the new problem.

The second problem under study is a celebrated optimization problem coming from network reliability analysis. We assume a graph G with perfect nodes and imperfect links, that fail independently with identical probability $\rho \in [0,1]$. The reliability $R_G(\rho)$, is the probability that the resulting subgraph has some spanning tree. Given a number of nodes and links, p and q, the goal is to find the (p,q)-graph that has the maximum reliability $R_G(\rho)$, uniformly in the compact set $\rho \in [0,1]$. In a first contribution, we exploit properties shared by all uniformly most-reliable graphs such as maximum connectivity and maximum Kirchhoff number, in order to build a novel GRASP/VND methodology. Our proposal finds the globally optimum solution under small cases, and it returns novel candidates of uniformly most-reliable graphs, such as Kantor-Möbius and Heawood graphs. We also offer a literature review, and a mathematical proof that the bipartite graph $K_{4,4}$ is uniformly most-reliable.

Finally, an abstract mathematical model of Stochastic Binary Systems (SBS) is also studied. It is a further generalization of network reliability models, where failures are modelled by a general logical function. A geometrical approximation of a logical function is offered, as well as a novel method to find reliability bounds for general SBS. This bounding method combines an algebraic duality, Markov inequality and Hahn-Banach separation theorem between convex and compact sets.

Keywords— Max Cut-Clique, Uniformly Most-Reliable Graph, Stochastic Binary System, GRASP, VND

List of publications issued from this thesis work

This thesis was written using a Swedish PhD style. The chapters are based on the following published and submitted papers:

1. "A GRASP/VND Heuristic for the Max Cut-Clique Problem", Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero and Luis Stábile. In Proccedings of the Fourth International Conference on Machine Learning, Optimization, and Data Science (LOD 2018), September 13-16, 2018, Volterra, Tuscany, Italy. Published by Springer in Lecture Notes in Computer Science series, Vol . 11331, ISBN: 978-3-030-13708-3, Pages 357–367.

2. "Complexity and Heuristics for the Max Cut-Clique Problem", Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero and Luis Stábile. In Proccedings of the Sixth International Conference on Variable Neighborhood Search (VNS 2018), October 4-7, 2018, Sithonia, Halkidiki, Greece. Published by Springer in Lecture Notes in Computer Science series, Vol. 11328, ISBN: 978-3-030-15843-9, Pages 28–40.

3. "Complexity and Heuristics for the Weighted Max Cut-Clique Problem", Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero and Luis Stábile. Submitted to International Transactions in Operational Research Journal published by Wiley Online Library.

4. "A Hybrid GRASP/VND Heuristic for the Design of Highly Reliable Networks", Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero and Luis Stábile. In Proccedings of the Eleventh International Workshop on Hybrid Metaheuristics (HM 2019), Juanary 16-18, 2019, Concepción, Chile. Published by Springer in Lecture Notes in Computer Science series, Vol. 11299, ISBN: 978-3-030-05983-5, Pages 78–92.

5. "Building Highly Reliable Networks with GRASP/VND Heuristics", Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero and Luis Stábile. In Proccedings of the Fifteenth International Conference on the Design of Reliable Communication Networks (DRCN 2019), March 19-21, 2019, Coimbra, Portugal. Published by IEEE 2019, ISBN 978-1-5386-8461-0, Pages 91–98.

6. "Analysis and Reliability of Separable Systems", Mathias Bourel, Héctor Cancela, Gustavo Guerberoff, Franco Robledo, Pablo Romero and Luis Stábile. Submitted to Networks Journal published by Wiley Online Library.

7. "Model Construction in Stochastic Binary Systems", Mathias Bourel, Héctor Cancela, Gustavo Guerberoff, Franco Robledo, Pablo Romero and Luis Stábile. Submitted to Eleventh International Workshop on Resilient Networks Design and Modeling (RNDM 2019), October 14-16, 2019, Nicosia, Cyprus.

Agradecimientos

El presente trabajo es resultado de un camino recorrido gracias a muchas personas que supieron guiarme, acompañarme, aconsejarme y apoyarme de forma incondicional, incluso en los momentos más difíciles.

En primer lugar quiero agradecer a mis tutores Franco Robledo, Pablo Romero y Mathias Bourel, por todas las horas que me dedicaron y por confiar en mí. Franco Robledo quién creyó, confió y guió desde desde el principio; quién supo despertar mi pasión por la investigación. No tengo más que palabras de agradecimiento hacia él. Pablo Romero, quién con sus apoyo incondicional, sabios consejos, sus meticulosas observaciones y oportunas correcciones ha sido un pilar fundamental en este trabajo y en mi formación como Doctor. Mathias Bourel, por su apoyo y consejos en cada uno de los trabajos.

Quiero reservar un espacio muy especial para agradecer a Eduardo Canale, co-autor en la mayoría de mis trabajos, quién sumó muchísimo en esta tesis y en mi formación. Muchas gracias por apoyarme siempre, por tu confianza, por las horas de trabajo dedicadas a mí, a reuniones y a discusiones grupales.

En segundo lugar quiero agradecer a Gustavo Guerberoff y a Héctor Cancela por el apoyo en los trabajos que compartimos.

A los revisores y al resto de tribunal, quienes me honran con la lectura de este trabajo.

Finalmente, pero no menos importante quiero agradecer a mi familia por el apoyo incondicional. A mis padres por las oportunidades que me dieron, a mi novia, mis hermanos y amigos por apoyarme siempre.

Acknowledgments

The present work is the result of a path traveled thanks to many people who knew how to guide me, accompany me, advise me and support me unconditionally, even in the most difficult moments.

First of all I want to thank my advisors Franco Robledo, Pablo Romero and Mathias Bourel, for all the hours shared and for trusting in me. Franco Robledo who believed in me and guided me from the beginning; who knew how to wake my passion for researching. I have nothing but thankful words to him. Pablo Romero, who with his unconditional support, wise advice, meticulous observations and timely corrections have been a fundamental pillar in this work and in my education as a doctor. Mathias Bourel, for his support and advice in each of the works.

I want to reserve a very special space to thank Eduardo Canale, coauthor in most of the works, who added a lot in this thesis and in my education. Thank you very much for always supporting me, for your confidence, for the hours of work dedicated to me, to meetings and to group discussions.

In the second place, I want to thank Gustavo Guerberoff and Héctor Cancela for the support in the works we shared.

To the reviewers and the rest of the court, who honored me by reading this work.

Last but not least I want to thank my family for the unconditional support. To my parents for the opportunities they gave me, my girlfriend, my brothers, and friends for always supporting me.

Chapter 1

Introduction

Two hard combinatorial problems are addressed in this thesis. The first one is known as the "Max Cut-Clique" and the second one coming from network reliability analysis. Finally, an abstract mathematical model of Stochastic Binary Systems (SBS) is also studied.

Searching for dense components in a network has long been attracting many researchers from different areas. Among those structures there is the concept of a clique, in which all elements are pairwise adjacent. This structure is expected to reveal a strongly related set of elements. A large number of applications involving cliques have been discussed in the literature. Some of those applications can be found in coding theory, fault diagnosis, computer vision, pattern recognition between others. Finding the maximum cardinality clique in G is known as the maximum clique (MC) problem.

Instead of searching for the largest size clique in the graph, we want a clique (of any size) with the largest number of edges incident to the nodes in the clique, excluding those within the clique. This problem has been introduced by (Martins, 2012), where formulations were proposed and their applicability to some real-world problems was shown.

Given an undirected graph $G = (V, E)$ and a clique C of G, the cut-clique is the set of edges running between C and $V \setminus C$, establishing the cut $(C, V \setminus C)$. The MCC in G is to find a clique with the largest number of edges in the neighborhood of the clique, also known as the maximum edge-neighborhood clique ($MENC$). We can generalize this problem considering the weights associated with each link. In this context we are interested in finding the clique $C \subset V$ such that the weighted-sum associated to each link shared between C and $V \setminus C$ is maximized. The weighted version of MCC is known as $MEWNC$ (Maximum Edge-Weight Neighborhood Clique).

Some of their applications can be found in Market Basket Analysis (MBA), sometimes known as *affinity analysis*. The goal of MBA is to identify non-obvious or counterintuitive relationships between groups of products, items, or categories. The determination of a set of items with a large correlation with others is a valuable tool in this context. The information obtained from MBA can have an important impact in the business strategy and operations. In the specific case of marketing, we can find valuable applications such as product placement, optimal product-line offering, personalized marketing campaigns and product promotions. In this thesis, we present algorithms for tackling this problem and compare the results with the state of the art.

The second part of this thesis refers to networks represented by graphs. Components can be either nodes or links which connect nodes. The study of the structure, the introduction of minimum levels of connectivity between their nodes, redundancy and resilience are main factors to avoid outages in case of a failure. Basically, the goal is to find the probability of correct operation of a system (Colbourn, 1999;

Beineke et al., 2012).

Several researchers from different fields of knowledge (mathematics, computer science, engineering), shaped the body of network reliability analysis, given the application and importance of the underlying models. A fundamental problem is to find the connectedness probability of a random graph, subject to link failures, called the *all-terminal reliability*. The scientific literature around this problem is vast; however, this problem is not fully understood yet. The corresponding practical problem is to connect p sites using q links in the *best* way, this is, to find the graph whose all-terminal reliability is maximum among all (p,q)-graphs. Such graphs are called *uniformly most-reliable graphs*.

The third part of this thesis refers to a more abstract setting of system reliability with stochastic binary systems (SBS). A SBS is a mathematical model of multi-component on-off system, where its components are subject to random failures. The number of feasible states for a system with N on-off components is 2^N, and an exhaustive list of all the states is computationally prohibitive. Recent works confirm that there is a special class of SBS called *separable system*, that accept an efficient representation, an the truth-table can be found by an inner product in the Euclidean space (Cancela et al., 2018).

1.1 Structure of the Thesis

This thesis follows the Swedish style, and it is organized into Three Parts. These parts have been ordered according to our chronological study. Part I contains the contributions related to the *MCC* and its weighted version *MEWNC*:

1. The computational complexity of both *MCC* and *MEWNC* is established.
2. A GRASP/VND methodology enriched with a Tabu Search is developed, where the main ingredients are novel local searches and a Restricted Candidate List that trades greediness for randomization in a multi-start fashion.
3. An exact Integer Linear Programming (ILP) formulation including bounds is proposed.
4. We offer bounds for both a globally optimal solution and the clique size are produced using elementary graph theory.
5. A fair comparison with respect to recent heuristics reveals that our proposal is competitive with state-of-the-art solutions

Part II presents the contributions on uniformly most-reliable graphs:

1. An exact VND that returns uniformly most-reliable graphs is presented.
2. A hybrid GRASP/VND heuristic is introduced in order to find graphs with high reliability. It trades quality for computational feasibility.
3. Novel networks that show high reliability and connectivity are found, as a result of our hybrid heuristic.
4. Our study confirms that our resulting graphs achieve the maximum tree-number (therefore, they are the only candidates of uniformly most-reliable graphs), and the maximum girth as well.

Finally, a more abstract setting of system reliability is presented in Part III:

1. A model construction methodology is proposed for the structure of stochastic binary systems.
2. A full reliability analysis and model construction is offered for the distinguished systems.
3. The hardness of reliability evaluation of separable systems is established.

1.2 Bibliography

Agrawal, R., Imieliński, T., and Swami, A. (1993). Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216.

Aguinis, H., Forcum, L. E., and Joo, H. (2013). Using market basket analysis in management research. *Journal of Management*, 39(7):1799–1824.

Amuthan, A. and Thilak, K. D. (2016). Survey on tabu search meta-heuristic optimization. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pages 1539–1543.

Bader, G. D. and Hogue, C. W. V. (2003). An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4:2.

Ball, M. O. (1980). Complexity of network reliability computations. *Networks*, 10(2):153–165.

Ball, M. O. (1986). Computational complexity of network reliability analysis: An overview. *IEEE Transactions on Reliability*, 35(3):230 –239.

Barrera, J., Cancela, H., and Moreno, E. (2015). Topological optimization of reliable networks under dependent failures. *Operations Research Letters*, 43(2):132 – 136.

Barta, J. and Montemanni (2012). Wireless multicasting under probabilistic node failures: a heuristic approach. *Optimization and Engineering*, 13(4):705–726.

Batagelj, V. and Mrvar, A. (2009). Pajek datasets (2006).

Bauer, D., Boesch, F., Suffel, C., and Slyke, R. V. (1987). On the validity of a reduction of reliable network design to a graph extremal problem. *IEEE Transactions on Circuits and Systems*, 34(12):1579–1581.

Beineke, L., Wilson, R., and Oellermann, O. (2012). *Topics in Structural Graph Theory*. Encyclopedia of Mathematics and its Applications. Cambridge University Press.

Biggs, N. (1993). *Algebraic Graph Theory*. Cambridge Mathematical Library. Cambridge University Press.

Boesch, F. T., Li, X., and Suffel, C. (1991). On the existence of uniformly optimally reliable networks. *Networks*, 21(2):181–194.

Brohée, S. and van Helden, J. (2006). Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7(1):488.

Bruinsma, G. and Bernasco, W. (2004). Criminal groups and transnational illegal markets. *Crime, Law and Social Change*, 41(1):79–94.

Canale, E., Cancela, H., Piccini, J., Robledo, F., Romero, P., Rubino, G., and Sartor, P. (2015a). Recursive variance reduction method in stochastic monotone binary systems. In *2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM)*, pages 135–141.

Canale, E., Cancela, H., Robledo, F., Romero, P., and Sartor, P. (2015b). Diameter constrained reliability: Complexity, distinguished topologies and asymptotic behavior. *Networks*, 66(4):296–305.

Canale, E., Cancela, H., Robledo, F., Romero, P., and Sartor, P. (2015c). Full complexity analysis of the diameter-constrained reliability. *International Transactions in Operational Research*, 22(5):811–821.

Canale, E., Romero, P., Rubino, G., and Warnes, X. (2016). Network utility problem and easy reliability polynomials. In *2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 79–84.

Cancela, H., Ferreira, G., Guerberoff, G., Robledo, F., and Romero, P. (2018). Building reliability bounds in stochastic binary systems. In *10th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 1–7.

Cancela, H., Guerberoff, G., Robledo, F., and Romero, P. G. (2018). Reliability maximization in stochastic binary systems. In *14th International Conference on the Design of Reliable Communication Networks (DRCN 2018)*, Paris, France.

Cancela, H. and Urquhart, M. E. (2002). Adapting rvr simulation techniques for residual connectedness network reliability models. *IEEE Transactions on Computers*, 51(4):439–443.

Cascio, W. F. and Aguinis, H. (2008). Research in industrial and organizational psychology from 1963 to 2007: Changes, choices, and trends. *Journal of Applied Psychology*, 93(5):1062–1081.

Colbourn, C. J. (1999). Reliability issues in telecommunications network planning. In *Telecommunications network planning, chapter 9*, pages 135–146. Kluwer Academic Publishers.

Conway, J. (1994). *A Course in Functional Analysis*. Graduate Texts in Mathematics. Springer New York.

Cook, S. A. (1971). The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, STOC '71, pages 151–158, New York, NY, USA. ACM.

Cramér, H. (1976). Half a century with probability theory: some personal recollections. *Annals of Probability*, 4:509–546.

Duarte, A., Mladenović, N., Sánchez-Oro, J., and Todosijević, R. (2016). *Variable Neighborhood Descent*, pages 1–27. Springer International Publishing, Cham.

Fishman, G. S. (1996). *Monte Carlo: Concepts, Algorithms and Applications*. Springer Verlag, New York, NY, USA.

Förster, J., Famili, I., Fu, P., Palsson, B. Ø., and Nielsen, J. (2003). Genome-scale reconstruction of the saccharomyces cerevisiae metabolic network. *Genome research*, 13(2):244–253.

Frederickson, G. N. and Ja'ja, J. (1982). On the relationship between the biconnectivity augmentation and travelling salesman problems. *Theoretical Computer Science*, 19(2):189–201.

Garey, M. R. and Johnson, D. S. (1979). *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, NY, USA.

Glover, F. and Laguna, M. (1997). *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA.

Hajek, B. and Zhu, J. (2010). The missing piece syndrome in peer-to-peer communication. In *2010 IEEE International Symposium on Information Theory*, pages 1748–1752.

Harary, F. (1962). The maximum connectivity of a graph. *Proceedings of the National Academy of Sciences of the United States of America*, 48(7):1142–1146.

Henzinger, M. and Lawrence, S. (2004). Extracting knowledge from the world wide web. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5186–5191.

Hüffner, F., Komusiewicz, C., Moser, H., and Niedermeier, R. (2008). Enumerating isolated cliques in synthetic and financial networks. In Yang, B., Du, D.-Z., and Wang, C. A., editors, *Combinatorial Optimization and Applications*, pages 405–416, Berlin, Heidelberg. Springer Berlin Heidelberg.

Jaeger, F., Vertigan, D. L., and Welsh, D. J. A. (1990). On the computational complexity of the jones and tutte polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, 108(1):35–53.

Jin, R., Wang, B., Wei, W., Zhang, X., Chen, X., Bar-Shalom, Y., and Willett, P. (2016). Detecting node failures in mobile wireless networks: A probabilistic approach. *IEEE Transactions on Mobile Computing*, 15(7):1647–1660.

Karp, R. M. (1972). Reducibility among combinatorial problems. In Miller, R. E. and Thatcher, J. W., editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press.

Kelmans, A. K. (1981). On graphs with randomly deleted edges. *Acta Mathematica Academiae Scientiarum Hungarica*, 37(1):77–88.

Kirchoff, G. (1847). Über die auflösung der gleichungen, auf welche man bei der untersuchung der linearen verteilung galvanischer ströme geführt wird. *Ann. Phys. Chem.*, 72:497–508.

Lee, H. W., Lee, K., and Modiano, E. (2014). Maximizing reliability in wdm networks through lightpath routing. *IEEE/ACM Transactions on Networking*, 22(4):1052–1066.

Levitin, G. (2006). *Computational Intelligence in Reliability Engineering: Evolutionary Techniques in Reliability Analysis and Optimization*. Studies in Computational Intelligence. Springer Berlin Heidelberg.

Lin, M. (2004). An efficient algorithm for computing the reliability of stochastic binary systems. *IEICE Transactions*, 87-D(3):745–750.

Markowetz, F. and Boutros, M. (2015). *Linking Phenotypes and Genotypes*. Cambridge Series in Systems Genetics. Cambridge University Press.

Martins, P. (2012). Cliques with maximum/minimum edge neighborhood and neighborhood density. *Computers & Operations Research*, 39(3):594–608.

Martins, P. and Gouveia, L. (2015). Solving the maximum edge-weight clique problem in sparse graphs with compact formulations. *EURO Journal on Computational Optimization*, 3:1–30.

Martins, P., Ladrón, A., and Ramalhinho, H. Maximum cut-clique problem: ILS heuristics and a data analysis application. *International Transactions in Operational Research*, 22(5):775–809.

Monma, C., Munson, B., and Pulleyblank, W. (1990). Minimum-weight two-connected spanning networks. *Mathematical Programming*, 46(1-3):153–171.

Petingi, L. and Rodriguez, J. (2001). Reliability of networks with delay constraints. In *Congressus Numerantium*, volume 152, pages 117–123.

Provan, S. J. and Ball, M. O. (1983). The Complexity of Counting Cuts and of Computing the Probability that a Graph is Connected. *SIAM Journal on Computing*, 12(4):777–788.

Pullan, W. (2008). Approximating the maximum vertex/edge weighted clique using local search. *Journal of Heuristics*, 14(2):117–134.

Rela, G., Robledo, F., and Romero, P. (2018). Petersen graph is uniformly most-reliable. In *Machine Learning, Optimization, and Big Data*, pages 426–435, Cham. Springer International Publishing.

Resende, M. G. and Ribeiro, C. C. (2016). *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*. Computational Science and Engineering. Springer-Verlag New York.

Romero, P. (2016). Duality in stochastic binary systems. In *2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 85–91.

Romero, P. (2017). Building uniformly most-reliable networks by iterative augmentation. In *9th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 1–7.

Rosenthal, A. (1977). Computing the reliability of complex networks. *SIAM Journal on Applied Mathematics*, 32(2):384–393.

Salhi, S. (2014). Handbook of metaheuristics (2nd edition). *Journal of the Operational Research Society*, 65(2):320–320.

Satyanarayana, A., Schoppmann, L., and Suffel, C. L. (1992). A reliability improving graph transformation with applications to network reliability. *Networks*, 22(2):209–216.

Song, X., Zhai, Z., Liu, Y., and Han, J. (2018). A stochastic approach for the reliability evaluation of multi-state systems with dependent components. *Reliability Engineering & System Safety*, 170(Supplement C):257 – 266.

Tan, P.-N., Steinbach, M., and Kumar, V. (2005). *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Viera, J. (2018). Búsqueda de grafos cúbicos de máxima confiabilidad. Master's thesis, Facultad de Ingeniería, Universidad de la República, Uruguay.

Wang, G. (1994). A proof of Boesch's conjecture. *Networks*, 24(5):277–284.

Part I

Max Cut-Clique Problem

Chapter 2

A GRASP/VND Heuristic for the Max Cut-Clique Problem

In this chapter, the complexity of the *MCC* is established. Considering the complexity proved for the *MCC* problem, which belongs to the class of $\mathcal{NP}$-Complete problems, we developed an Heuristic GRASP/VND methodology enriched with Tabu Search. Finally a fair comparison that our approach is comparative with state-of-the art solutions.

A GRASP/VND Heuristic for the Max Cut-Clique Problem

Mathias Bourel, Eduardo Canale,
Franco Robledo, Pablo Romero, and Luis Stábile

Instituto de Matemática y Estadística, IMERL
Facultad de Ingeniería - Universidad de la República
Montevideo Uruguay
(mbourel,canale,frobledo,promero,lstabile)@fing.edu.uy,

Abstract. In Market Basket Analysis, the goal is to understand the human behavior in order to maximize sales. An evident behavior is to buy correlated items. As a consequence, the determination of a set of items with a large correlation with others is a valuable tool for Market Basket Analysis.

In this paper we address a combinatorial optimization problem that formalizes the previous application. Given a simple graph $\mathcal{G} = (V, E)$ (where the nodes are items and links represent correlation), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized. This problem is known in the literature as Max Cut-Clique (MCC).

The contributions of this paper are three-fold. First, the computational complexity of the MCC is established. Second, a full GRASP/VND methodology enriched with a Tabu Search is here developed, where the main ingredients are novel local searches and a Restricted Candidate List that trades greediness for randomization in a multi-start fashion. A Tabu Search is also included in order to avoid locally optimum solutions. Finally, a fair comparison with respect to recent heuristics reveals that our proposal is competitive with state-of-the-art solutions.

Keywords: Market Basket Analysis, Combinatorial Optimization, Max Cut-Clique, Metaheuristics

1 Motivation

There is a serious disconnection between the knowledge that academics are producing and the knowledge that practitioners are consuming [7]. A bridge between the science-practice division can be found in Market Basket Analysis (MBA), sometimes known as *affinity analysis* [2]. In synthesis, MBA is a Data Mining technique [1, 19] originated in the field of marketing. It has recent applications to other fields, such as bioinformatics [4, 5], WWW networks [12], criminal networks [6] and financial networks [13]. The goal of MBA is to

identify non-obvious or counterintuitive relationships between groups of products, items, or categories.

The information obtained from MBA can have an important impact in the business strategy and operations. In the specific case of marketing, we can find valuable applications such as product placement, optimal product-line offering, personalized marketing campaigns and product promotions. The analysis is commonly supported by Machine Learning (pattern matching, clustering, feature extraction, statistics), Optimization and Logical rules for association.

This work is focused on a specific combinatorial optimization methodology to assist product placement; however, related applications could be found. The problem under study is called Max Cut-Clique (MCC), and it was introduced by P. Martins [15]. Given a simple graph $\mathcal{G} = (V, E)$ (where the nodes are items and links represent correlation), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized. The MCC has an evident application to product-placement. For instance, the manager of a supermarket must decide how to locate the different items in the different compartments. In a first stage, it is essential to determine the correlation between the different pairs of items, for psychological/attractive reasons. Then, the priceless/basic products (bread, rice, milk and others) could be hidden on the back, in order to give the opportunity for other products in a large corridor (and candies should be at hand by kids as well). Observe that the MCC appears in the first stage, while marketing/psychological aspects play a key role in a second stage for product-placement in a supermarket.

In [15], the author states that the MCC is presumably hard, since related problems such as $MAX-CUT$ and $MAX-CLIQUE$ are both $\mathcal{NP}$-Complete. To the best of our knowledge, there is no formal proof available for the hardness of the MCC in the published scientific literature. Nevertheless, the MCC is systematically addressed by the scientific community with metaheuristics and exact solvers that run in exponential time.

A recent work in the field develops an Iterated Local Search for the MCC [16]. As far as we know, this work belongs to the state-of-the-art techniques for the MCC. The authors find optimal solutions for most instances under study, and suggest a rich number of applications.

The contributions of this paper can be summarized in the following items:

1. The $\mathcal{NP}$-Completeness of the MCC is established (Section 2).
2. A hybrid GRASP/VND heuristic enriched with Tabu Search is developed to address the MCC (Section 3).
3. A fair comparison with a state-of-the-art heuristic is presented using DIMACS benchmark (Section 4).

2 Computational Complexity

The cornerstone in computational complexity is Cook's Theorem [8] and Karp reducibility among combinatorial problems [14].

Stephen Cook formally proved that the joint satisfiability of an input set of clauses in disjunctive form is the first $\mathcal{NP}$-Complete decision problem [8]. Furthermore, he provided a systematic procedure to prove that a certain problem is $\mathcal{NP}$-Complete. Specifically, it suffices to prove that the decision problem belongs to set $\mathcal{NP}$, and that it is at least as hard as an $\mathcal{NP}$-Complete problem. Richard Karp followed this hint, and presented the first 21 combinatorial problems that belong to this class [14]. In particular, $MAX - CLIQUE$ belongs to this list. The reader is invited to consult an authoritative book in Complexity Theory, which has a larger list of $\mathcal{NP}$-Complete problems and a rich number of bibliographic references [10].

Here, we formally prove that the MCC is at least as hard as $MAX - CLIQUE$. Let us denote $|\mathcal{C}|$ the cardinality of a clique $\mathcal{C}$, and $\delta(\mathcal{C})$ denotes the corresponding cutset induced by the clique (or the set) $\mathcal{C}$.

Definition 1 (MAX-CLIQUE).
GIVEN: a simple graph $G = (V, E)$ and a real number K.
QUESTION: is there a clique $\mathcal{C} \subseteq V$ such that $|\mathcal{C}| \geq K$?

For convenience, we describe MCC as a decision problem:

Definition 2 (MCC).
GIVEN: a simple graph $G = (V, E)$ and a real number K.
QUESTION: is there a clique $\mathcal{C} \subseteq G$ such that $|\delta(\mathcal{C})| \geq K$?

Theorem 1. *The MCC belongs to the class of $\mathcal{NP}$-Complete problems.*

Proof. We prove that the MCC is at least as hard as $MAX - CLIQUE$. Consider a simple graph $G = (V, E)$ with order $n = |V|$ and size $m = |E|$. Let us connect a large number of M hanging nodes, to every single node $v \in V$. The resulting graph is called H (see Figure 1 for an example). If we find a polynomial-time algorithm for MCC, then we can produce the max cut-clique in H. But observe that the Max Cut-Clique $\mathcal{C}$ in H cannot include hanging nodes, thus it must belong entirely to G. If a clique $\mathcal{C}$ has cardinality c, then the clique-cut has precisely $c \times M$ hanging nodes. By construction, the cut-clique must maximize the number of hanging nodes, if we choose $M \geq m$. As a consequence, c must be the $MAX - CLIQUE$. We proved that the MCC is at least as hard as $MAX - CLIQUE$, as desired. Since MCC belongs to the set of $\mathcal{NP}$ Decision problems, it belongs to the $\mathcal{NP}$-Complete class. ■

Theorem 1 promotes the development of heuristics in order to address the MCC.

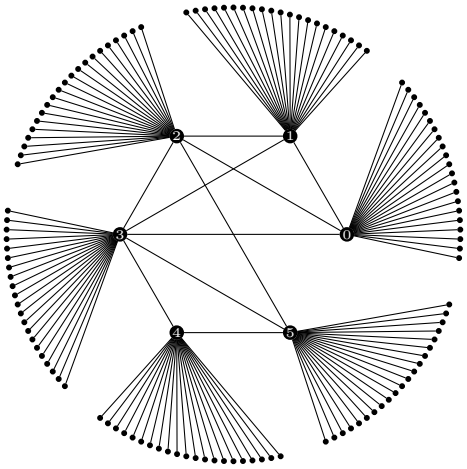


Fig. 1. Construction of H with $M = 21$ hanging nodes.

3 Methodology

GRASP and Tabu Search are well known metaheuristics that have been successfully used to solve many hard combinatorial optimization problems. GRASP is an iterative multi-start process which operates in two phases [17]. In the Construction Phase a feasible solution is built whose neighborhood is then explored in the Local Search Phase. Tabu Search [11, 3] is a strategy to prevent local search algorithms getting trapped in locally optimal solutions. A penalization mechanism called Tabu List is considered to avoid returning to previously visited solutions. For a complete description of these methods the reader is referred to the works of Glover and Laguna [11] and Resende and Ribeiro [17]. The reader is invited to consult the comprehensive Handbook of Metaheuristic for further information [18].

Here, we develop a GRASP/VND methodology enriched with Tabu Search in order to avoid getting trapped in previous visited solutions. In the following, the pseudocode of our Hybrid Metaheuristic (HM) for the Max Cut-Clique is presented. It follows the traditional two-phase GRASP template enriched with a Variable Neighborhood Descent (Lines 4-5). A Tabu Search strategy is included in order to enhance feasible solutions. The tabu list $\mathcal{T}$ stores tabu nodes (Line 2), discarding previous solutions. Essentially, the most frequent nodes involved in all solutions after the second phase of Variable Neighborhood Descent (VND) are not considered for further solutions during θ iterations, whenever we reach θ^{max} consecutive iterations without improvement. The most frequent nodes are selected if they appear more than ϕ times since the last tabu list refresh. The real numbers ϕ and θ are uniformly chosen at random in the interval $[1, \theta^{max}]$, being θ^{max} a parameter of the algorithm. The specific GRASP phases for the MCC are described in detail in the following subsections.

Algorithm 1 HM PSEUDOCODE

Input: α, θ^{max}, *maxIter*, $\mathcal{G}$
Output: $\mathcal{C}^*$

1: $\mathcal{C}^* \leftarrow \emptyset$
2: $\mathcal{T} \leftarrow \emptyset$
3: **for** *iter = 1* **to** *maxIter* **do**
4: $\quad \mathcal{C} \leftarrow$ CLIQUE*(α, $\mathcal{T}$, $\mathcal{G}$)*
5: $\quad \mathcal{C} \leftarrow$ VND*($\mathcal{C}$, $\mathcal{T}$, $\mathcal{G}$)*
6: $\quad \mathcal{T} \leftarrow$ UPDATE*($\mathcal{T}$, θ^{max}, $\mathcal{C}$)* ▷ Tabu List
7: $\quad$ **if** $|E'(\mathcal{C})| > |E'(\mathcal{C}^*)|$ **then**
8: $\quad\quad \mathcal{C}^* \leftarrow \mathcal{C}$
9: **return** $\mathcal{C}^*$

3.1 Construction Phase - *Clique*

The construction phase of the proposed algorithm is depicted in Algorithm 2. Let us denote by $\mathcal{C}$ the clique under construction, $\delta(U)$ and $\Delta(U)$ the minimum and maximum degree of the node-set U. The clique $\mathcal{C}$ is initially empty (Line 1), and a multi-start process is considered (Line 2). A Restricted Candidate List, RCL, is defined in Line 3. Observe that the RCL includes nodes with the highest degree, and α trades greediness for randomization. During the *While* loop of Lines 4-11, a singleton $\{i\}$ is uniformly picked from the RCL (Line 5), and the maximum clique $\mathcal{C}'$ is built using all the nodes from the set $\mathcal{C} \cup \{i\}$ (see Line 6). The best solution is updated if necessary (Lines 7-8). Observe that the process is finished only if we meet $MAX\_ATTEMPTS$ without improvement (Lines 9-11). The reader can appreciate that the output $\mathcal{C}$ is the best feasible clique during the whole process (Line 12).

Algorithm 2 CLIQUE

Input: α, $\mathcal{T}$, $\mathcal{G}$
Output: $\mathcal{C}$

1: $\mathcal{C} \leftarrow \emptyset$
2: *improving* = MAX_ATTEMPTS
3: $RCL \leftarrow \{v \in V - \mathcal{C} : |E'(v)| \geq \Delta(V - \mathcal{C}) - \alpha(\Delta(V - \mathcal{C}) - \delta(V - \mathcal{C}))\}$
4: **while** *improving* > 0 **do**
5: $\quad i \leftarrow selectRandom(RCL)$
6: $\quad \mathcal{C}' \leftarrow [\mathcal{C} \cap N(i)] \cup \{i\}$
7: $\quad$ **if** $|E'(\mathcal{C}')| > |E'(\mathcal{C})|$ **then**
8: $\quad\quad \mathcal{C} \leftarrow \mathcal{C}'$
9: $\quad\quad$ *improving* $\leftarrow$ MAX_ATTEMPTS
10: $\quad$ **else**
11: $\quad\quad$ *improving* $\leftarrow$ *improving* $- 1$
12: **return** $\mathcal{C}$

3.2 Local Search Phase - *VND*

The goal is to combine a rich diversity of neighborhoods in order to obtain an output that is locally optimum solution for every feasible neighborhood. Five neighborhood structures are considered to build a VND [9].

- **Remove**: a singleton $\{i\}$ is removed from a clique $\mathcal{C}$.
- **Add**: a singleton $\{i\}$ is added from a clique $\mathcal{C}$.
- **Swap**: if we find $j \notin \mathcal{C}$ such that $\mathcal{C} - \{i\} \subseteq N(j)$, we can include j in the clique and delete i (swap i and j).
- **Cone**: generalization of Swap for multiple nodes. The clique $\mathcal{C}$ is replaced by $\mathcal{C} \cup \{i\} - \mathcal{A}$, being $\mathcal{A}$ the nodes from $\mathcal{C}$ that are non-adjacent to i.
- **Aspiration**: this movement offers the opportunity of nodes belonging to the Tabu List to be added.

The previous neighborhoods take effect whenever the resulting cut-clique is increased. It is worth to remark that **Add**, **Swap**, and **Aspiration** are taken from a previous ILS [16]. However, our VND is enriched with 2 additional neighborhood structures, named **Remove** and **Cone**. Observe that the Tabu list works during the potential additions during **Add**, **Swap** and **Cone**. On the other hand, **Aspiration** provides diversification with an *opportunistic unchoking* process: it picks nodes from the Tabu List instead.

For the remaining four local searches, there is an efficient way to determine whether there is an improvement with respect to some neighbor-set. Specifically, the Test Lemmas 1 to 4 are useful to determine the improvements for **Remove**, **Add**, **Swap** and **Cone** movements, respectively. We call Aspiration Test to Lemma 2 but applied in a different domain (specifically, the candidate nodes must belong to the Tabu List).

Lemma 1 (Remove). $|\delta(\mathcal{C} - \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| < 2(|\mathcal{C}| - 1)$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \{i\})| &= |\delta(\mathcal{C})| + |\mathcal{C}| - 1 - (|\delta(i)| - (|\mathcal{C}| - 1)) \\ &= |\delta(\mathcal{C})| + |\mathcal{C}| - 1 - |\delta(i)| + |\mathcal{C}| - 1 \\ &= \delta(\mathcal{C})| + 2(|\mathcal{C}| - 1) - |\delta(i)| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $2(|\mathcal{C}| - 1) - |\delta(i)| > 0$. ■

Lemma 2 (Add). $|\delta(\mathcal{C} \cup \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > 2|\mathcal{C}|$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} \cup \{i\})| &= |\delta(\mathcal{C})| - |\mathcal{C}| + |\delta(i)| - |\mathcal{C}| \\ &= |\delta(\mathcal{C})| + |\delta(i)| - 2|\mathcal{C}| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > 2|\mathcal{C}|$. ■

Lemma 3 (Swap). $|\delta(\mathcal{C} - \{j\} \cup \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > |\delta(j)|$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \{j\} \cup \{i\})| &= |\delta(\mathcal{C})| - |\delta(j)| + 2(|\mathcal{C}| - 1) + |\delta(i)| - 2(|\mathcal{C}| - 1) \\ &= |\delta(\mathcal{C})| - |\delta(j) + |\delta(i)| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > |\delta(j)|$. ■

Lemma 4 (Cone). $|\delta(\mathcal{C} - \mathcal{A} \cup \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > |\delta(\mathcal{A})| - 2|\mathcal{C} - \mathcal{A}|(|\mathcal{A}| - 1)$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \mathcal{A} \cup \{i\})| &= |\delta(\mathcal{C})| + |\mathcal{A}||\mathcal{C} - \mathcal{A}| - (|\delta(\mathcal{A})| - |\mathcal{A}||\mathcal{C} - \mathcal{A}|) - 2|\mathcal{C} - \mathcal{A}| + |\delta(i)| \\ &= |\delta(\mathcal{C})| + 2|\mathcal{A}||\mathcal{C} - \mathcal{A}| - |\delta(\mathcal{A})| - 2|\mathcal{C} - \mathcal{A}| + |\delta(i)| \\ &= |\delta(\mathcal{C})| + 2|\mathcal{C} - \mathcal{A}|(|\mathcal{A}| - 1) - |\delta(\mathcal{A})| + |\delta(i)||\delta(\mathcal{C} - \mathcal{A} \cup \{i\})| \\ &> |\delta(\mathcal{C})| \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > |\delta(\mathcal{A})| - 2|\mathcal{C} - \mathcal{A}|(|\mathcal{A}| - 1)$. ■

The Flow Diagram of our VND is presented in Figure 2. The ordered sequence of local searches are **Remove**, **Add**, **Swap**, **Cone** and **Aspiration** moves. Once an improvement is obtained, the process restarts from the beginning. Observe that, in the output, a locally optimum solution under all neighborhood structures is met.

4 Computational Results

In order to test the performance of the algorithm, a fair comparison with respect to an Iterated Local Search solution [16] is carried out using DIMACS benchmark. The test was executed on an Intel Core i7, 2.4 GHz, 8GB RAM.

Table 1 reports the performance of our HM algorithm for each instance [1]. All instances were tested using 100 runs with $\alpha = \frac{1}{2}$, $MAX\_ATTEMPTS = \lfloor \frac{|V|}{10} \rfloor$, $\theta^{max} = 10$. The values remarked using bold letters from column $|E'(\mathcal{C})|$ indicate that the best solution known was reached according to [16].

Following the terminology, *max_iter* represents the number of iterations considered in the algorithm, $|E'(\mathcal{C})|$, $|\mathcal{C}|$ and $Time$ represent maximum cut-clique size found, best solution and the CPU time for the Best solution found. The same columns are reported for an averaging over 100 runs.

[1] All the scripts are available at the following URL: `https://www.fing.edu.uy/~lstabile/mcc-octave-source.zip`

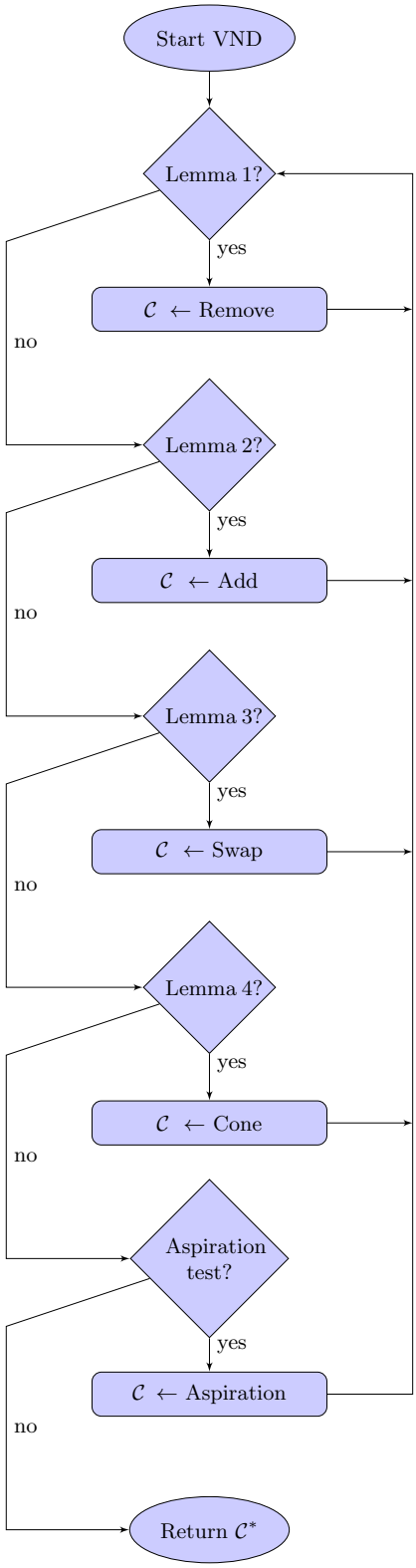


Fig. 2. Flow Diagram for the Local Search Phase - VND.

Instances			Parameters	Best			Average	
name	n	Density	max_iter	$\lvert E'(\mathcal{C})\rvert$	$\lvert\mathcal{C}\rvert$	Time (s)	$\lvert E'(\mathcal{C})\rvert$	Time (s)
c-fat200-1	200	0.071	10	**81**	9	0.1	81	0.3
c-fat200-2	200	0.163	10	**306**	17	0.5	306	0.8
c-fat200-5	200	0.426	10	**1892**	43	3	1892	4.9
c-fat500-1	500	0.036	10	**110**	10	0.5	110	2.4
c-fat500-2	500	0.073	10	**380**	19	3	380	5.8
c-fat500-5	500	0.186	10	**2304**	48	10	2304	10.8
c-fat500-10	500	0.374	10	**8930**	94	38	8930	65
p_hat300-1	300	0.244	100	**789**	8	129	787	905
p_hat300-2	300	0.489	100	**4637**	25	8	4636	3659
p_hat300-3	300	0.744	1000	**7740**	36	469	7556	3992
p_hat500-1	500	0.253	100	**1621**	9	13	1621	694
p_hat500-2	500	0.505	100	**11539**	36	16	11401	723
p_hat500-3	500	0.752	1000	**18859**	50	679	18855	723
p_hat700-1	700	0.249	100	**2606**	11	305	2602	439
p_hat700-2	700	0.498	1000	**20425**	44	79	20425	839
p_hat700-3	700	0.748	1000	**33480**	62	945	33468	1807
p_hat1000-1	1000	0.245	1000	**3556**	10	216	3556	355
p_hat1000-2	1000	0.490	10000	**31174**	46	2124	31174	2538
p_hat1000-3	1000	0.744	10000	**51259**	65	2687	53256	3584
p_hat1500-1	1500	0.253	1000	**6018**	11	399	6018	904
p_hat1500-2	1500	0.506	10000	**67486**	65	2482	67486	2942
p_hat1500-3	1500	0.754	10000	**112873**	94	1174	112872	23162
keller4	171	0.649	100	**1140**	11	9	1140	11
keller5	776	0.752	10000	**15184**	27	1956	15183	1167
keller6	3361	0.818	100000	**159608**	59	26362	158423	321731
c125_9	125	0.899	1000	**2766**	34	102	2766	253
c250_9	250	0.899	1000	**8123**	44	426	8123	831
c500_9	500	0.901	10000	**22691**	57	2354	22652	4469
c1000_9	1000	0.901	10000	**57149**	68	3924	56038	4125
c2000_5	2000	0.500	10000	**16106**	16	23472	16082	23472
c2000_9	2000	0.900	50000	**136769**	79	37472	135001	45472
c4000_5	4000	0.500	50000	**36174**	18	31196	35891	38119
MANN_a9	45	0.927	1000	**412**	16	4	412	145
MANN_a27	378	0.990	10000	**31284**	126	309	31244	548
MANN_a45	1035	0.996	50000	**236406**	344	46881	235072	52112
MANN_a81	3321	0.999	50000	**2436894**	1098	73213	2433624	96743

Table 1. Results of the algorithm for the *MCC* problem

The reader can appreciate that our HM algorithm meets the best solution known so far in all cases. On the one hand, HM is a more powerful strategy than ILS, since the local search from the latter are completely included in the former. On the other, the computational effort is increased using HM. Even though a globally optimum is not formally proved for some instances, the null gap between ILS and our solution suggests an evidence of optimality.

The results described in this section reflect that our GRASP/VND methodology is competitive with state-of-the-art solutions for the MCC. We underscore the simplicity of implementation conducted by simple building blocks (solution construction procedures and local search methods).

5 Conclusions and Trends for Future Work

Several business models can be represented by Market Basket Analysis (MBA). A relevant marketing approach is to find a subset of items that are strongly correlated with the others. This intuition is formalized by means of a combinatorial optimization problem, called Max Cut-Clique (MCC). In this paper the $\mathcal{NP}$-Completeness of MCC is established. Then, a GRASP/VND methodology enriched with Tabu Search is developed to address the MCC. A fair comparison confirms that our approach is competitive with state-of-the art solutions.

As future work, we want to implement our solution into a real-life product-placement scenario. In a first stage, we need historical information to determine the links between pairs of items. Finally, the physical location of the items must be determined using a complementary geometrical problem with constraints. The solution could consider multi-constrained clustering in order to include categories for the items, or other Machine Learning techniques to determine profiles for the customers, according to the product under study. After the real implementation, the feedback of sales in a period is a valuable metric of success.

6 Acknowledgements

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos*. We would like to thank the reviewers for their insightful comments that simplified the readability of this paper.

References

1. Rakesh Agrawal, Tomasz Imieliński, and Arun Swami. Mining association rules between sets of items in large databases. *SIGMOD Rec.*, 22(2):207–216, June 1993.
2. Herman Aguinis, Lura E. Forcum, and Harry Joo. Using market basket analysis in management research. *Journal of Management*, 39(7):1799–1824, 2013.
3. A. Amuthan and K. Deepa Thilak. Survey on tabu search meta-heuristic optimization. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pages 1539–1543, Oct 2016.

4. Gary D. Bader and Christopher W. V. Hogue. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics*, 4:2, 2003.
5. Sylvain Brohée and Jacques van Helden. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics*, 7(1):488, Nov 2006.
6. Gerben Bruinsma and Wim Bernasco. Criminal groups and transnational illegal markets. *Crime, Law and Social Change*, 41(1):79–94, Feb 2004.
7. Wayne F. Cascio and Herman Aguinis. Research in industrial and organizational psychology from 1963 to 2007: Changes, choices, and trends. *Journal of Applied Psychology*, 93(5):1062–1081, 2008.
8. Stephen A. Cook. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, STOC '71, pages 151–158, New York, NY, USA, 1971. ACM.
9. Abraham Duarte, Nenad Mladenović, Jesús Sánchez-Oro, and Raca Todosijević. *Variable Neighborhood Descent*, pages 1–27. Springer International Publishing, Cham, 2016.
10. Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, NY, USA, 1979.
11. Fred Glover and Manuel Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA, 1997.
12. Monika Henzinger and Steve Lawrence. Extracting knowledge from the world wide web. *Proceedings of the National Academy of Sciences*, 101(suppl 1):5186–5191, 2004.
13. Falk Hüffner, Christian Komusiewicz, Hannes Moser, and Rolf Niedermeier. Enumerating isolated cliques in synthetic and financial networks. In Boting Yang, Ding-Zhu Du, and Cao An Wang, editors, *Combinatorial Optimization and Applications*, pages 405–416, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg.
14. R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.
15. Pedro Martins. Cliques with maximum/minimum edge neighborhood and neighborhood density. 39:594–608, 03 2012.
16. Pedro Martins, Antonio Ladrón, and Helena Ramalhinho. Maximum cut-clique problem: ILS heuristics and a data analysis application. *International Transactions in Operational Research*, 22(5):775–809, 2014.
17. Mauricio G.C. Resende and Celso C. Ribeiro. *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*. Computational Science and Engineering. Springer-Verlag New York, 2016.
18. Said Salhi. Handbook of metaheuristics (2nd edition). *Journal of the Operational Research Society*, 65(2):320–320, Feb 2014.
19. Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 2005.

Chapter 3

Complexity and Heuristics for the Max Cut-Clique Problem

In this Chapter, the $\mathcal{NP}$-Completeness of *MCC* promotes the development of heuristics and bounds. As a consequence, we offered bounds for both the globally optimum solution and the size of the minimum cardinality clique with maximum cut. Additionally, an exact Integer Linear Programming (ILP) formulation for the problem is offered. A more extensive experimental analysis is carried out including the results of the exact method.

Complexity and Heuristics for the Max Cut-Clique Problem

Mathias Bourel, Eduardo Canale,
Franco Robledo, Pablo Romero, and Luis Stábile

Instituto de Matemática y Estadística, IMERL
Facultad de Ingeniería - Universidad de la República
Montevideo Uruguay
(mbourel,canale,frobledo,promero,lstabile)@fing.edu.uy,

Abstract. In this paper we address a metaheuristic for an combinatorial optimization problem. For any given graph $\mathcal{G} = (V, E)$ (where the nodes represent items and edges correlations), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized. This problem is known in the literature as the Max Cut-Clique (MCC).

The contributions of this paper are three-fold. First, the complexity of the MCC is established, and we offer bounds for the MCC using elementary graph theory. Second, an exact Integer Linear Programming (ILP) formulation for the MCC is offered. Third, a full GRASP/VND methodology enriched with a Tabu Search is here developed, where the main ingredients are novel local searches and a Restricted Candidate List that trades greediness for randomization in a multi-start fashion. A dynamic Tabu list considers a bounding technique based on the previous analysis.

Finally, a fair comparison between our hybrid algorithm and the globally optimum solution using the ILP formulation confirms that the globally optimum solution is found by our heuristic for graphs with hundreds of nodes, but more efficiently in terms of time and memory requirements.

Keywords: Combinatorial Optimization Problem, Max Cut-Clique, ILP, GRASP, VND, Tabu Search.

1 Motivation

The MCC has an evident application to product-placement in Market Basket Analysis (MBA), sometimes known as *affinity analysis* [1]. For instance, the manager of a supermarket must decide how to locate the different items in the different compartments. In a first stage, it is essential to determine the correlation between the different pairs of items, for psychological/attractive reasons. Then, the priceless/basic products (bread, rice, milk and others) could be hidden on the back, in order to give the opportunity for other products in a large corridor (and

candies should be at hand by kids as well). Observe that the MCC appears in the first stage, while marketing/psychological aspects play a key role in a second stage for product-placement in a supermarket.

This work is focused on a specific combinatorial optimization methodology to assist product placement; however, related applications could be found. The problem under study is called Max Cut-Clique (MCC), and it was introduced by P. Martins [5]. For any given graph $\mathcal{G} = (V, E)$ (where the nodes are items and links represent correlation), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized.

In [5], the author states that the MCC is presumably hard, since related problems such as *MAX-CUT* and *MAX-CLIQUE* are both $\mathcal{NP}$-Complete. To the best of our knowledge, there is no formal proof available for the hardness of the MCC in the published scientific literature. Nevertheless, the MCC is systematically addressed by the scientific community with metaheuristics and exact solvers that run in exponential time.

A recent work in the field develops an Iterated Local Search for the MCC [6]. As far as we know, this work belongs to the state-of-the-art techniques for the MCC. The authors find optimal solutions for most instances under study, and suggest a rich number of applications.

The contributions of this paper can be summarized in the following items:

1. The $\mathcal{NP}$-Completeness of MCC is established (Subsection 2.1).
2. Bounds for both the globally optimum solution and the clique size are produced (Subsection 2.2).
3. A hybrid GRASP/VND heuristic enriched with Tabu Search is developed to address the MCC (Section 3).
4. An exact Integer Linear Programming (ILP) formulation for the MCC is proposed (Section 4).
5. The performance of our approach is studied (Section 5).
6. A discussion of applications for product-placement is included (Section 6).

2 Analysis and Complexity

In this section, the computational complexity for the MCC is established. We formally prove that the corresponding decision version for the MCC belongs to the class of $\mathcal{NP}$-Complete decision problems (Subsection 2.1). Then, we find bounds for the MCC using elementary graph theory (Subsection 2.2).

It is worth to remark that the hardness promotes the development of heuristics, and these bounds will enrich our GRASP/VND heuristic with a dynamic Tabu List.

2.1 Complexity

We formally prove that the MCC is at least as hard as *MAX-CLIQUE*. First, we describe both decision problems and the decision versions for the MCC:

Definition 1 (*MAX-CLIQUE*).
GIVEN: a graph $G = (V, E)$ and a real number K.
QUESTION: is there a clique $\mathcal{C} \subseteq V$ such that $|\mathcal{C}| \geq K$?

For convenience, we describe the MCC as a decision problem. Let us denote $\delta(\mathcal{C})$ to the cut produced by a node-set $\mathcal{C}$, or the objective value for the MCC whenever $\mathcal{C}$ is a clique.

Definition 2 (*MCC*).
GIVEN: a graph $G = (V, E)$ and a real number K.
QUESTION: is there a clique $\mathcal{C} \subseteq G$ such that $|\delta(\mathcal{C})| \geq K$?

Theorem 1. *The MCC belongs to the class of $\mathcal{NP}$-Complete problems.*

Proof. We prove that the MCC is at least as hard as *MAX-CLIQUE*. Consider a simple graph $G = (V, E)$ with order $n = |V|$ and size $m = |E|$. Let us connect m leaf-nodes hanging to every single node $v \in V$ (observe that there are $m \times n$ such nodes). The resulting graph is called H. If we find a polynomial-time algorithm for MCC, then we can produce the max cut-clique in H. But observe that the max cut-clique $\mathcal{C}$ in H must belong to G. If $\mathcal{C}$ has cardinality c, then the cut-clique has precisely $c \times m$ hanging nodes. By construction, the cut-clique must maximize the number of hanging nodes, since the whole size $|E| = m$ is added to the cut by a single addition of a node in the clique. As a consequence, c must be the *MAX-CLIQUE*. We proved that the MCC is at least as hard as *MAX-CLIQUE*, as desired. Since MCC belongs to the set of $\mathcal{NP}$ decision problems, it belongs to the $\mathcal{NP}$-Complete class. ■

2.2 Bounds for *MCC*

Observe that the globally optimum for the MCC could be attained by more than one clique. Let us denote by $\mathcal{C}_{min}$ the minimum cardinality clique such that $|\delta(\mathcal{C}_{min})| = OPT$, the optimal value for the MCC, and $c_{min} = |\mathcal{C}_{min}|$.

Definition 3. *A finite sequence $\{a_i\}_{i=0}^n$ is* strictly unimodal *if there exists some index $k_0 \in \{0, \ldots, n\}$ such that $a_0 < a_1 < \cdots < a_{k_0}$ and $a_{k_0} \geq a_{k_0+1} > a_{k_0+2} > \cdots > a_n$.*

Lemma 1. *Consider a connected graph G with degree-sequence $(\delta_1, \ldots, \delta_n)$, where for convenience we consider $\delta_1 \leq \delta_2 \leq \cdots \leq \delta_n$. Then, the following finite sequence $\{f(k)\}_{k=0}^n$ is strictly unimodal, where*

$$f(k) = -k(k-1) + \sum_{i=1}^{k} \delta_{n-i+1}, \, \forall k \in \{0, 1, \ldots, n\}. \tag{1}$$

Proof. The difference between consecutive terms, $\Delta_k = f(k) - f(k-1)$, is:

$$\Delta_k = -k(k-1) + \sum_{i=1}^{k} \delta_{n-i+1} + (k-1)(k-2) - \sum_{i=1}^{k-1} \delta_{n-i+1} = -2(k-1) + \delta_{n-k+1}.$$

Since $-2(k-1)$ and δ_{n-k+1} are monotonically decreasing sequences, being the former strictly decreasing, $\{\Delta_k\}_{k\geq 0}$ must be strictly decreasing as well. Furthermore, since $\Delta_1 = f(1) - f(0) = \delta_n > 0$ and $\Delta_n = -2(n-1) + \delta_1 < 0$, there exists some index k_0 such that: $f(0) < f(1) < \cdots < f(k_0)$ and $f(k_0) \geq f(k_0+1) > f(k_0+2) > \cdots > f(n)$, as desired. ■

Lemma 2. *The following inequalities hold for any clique $\mathcal{C}$:*

$$-|\mathcal{C}|(|\mathcal{C}|-1) + \sum_{i=1}^{|\mathcal{C}|} \delta_i \leq |\delta(\mathcal{C})| \leq f(|\mathcal{C}|) \tag{2}$$

Proof. The sum $|\delta(\mathcal{C})| + |\mathcal{C}|(|\mathcal{C}|-1) = \sum_{v_i \in \mathcal{C}} \delta_i$ is greater (smaller) than the sum of the $|\mathcal{C}|$ smallest (greatest) degrees. ■

In the following, we will provide an upper-bound for OPT, the globally optimum value for the MCC, and bounds for the size c_{min} of the minimum cardinality clique $\mathcal{C}_{min}$, in terms of the auxiliary sequence $\{f(k)\}_{k=0}^{n}$.

Theorem 2 (*Upper-Bound for* MCC).
If OPT denotes the optimal value for the MCC and f is maximized at k_0, then $OPT \leq f(k_0)$.

Proof. If we are given an arbitrary clique $\mathcal{C}$, the incident edges to some $v \in \mathcal{C}$ either belong to the clique or to the cut. Then:

$$|\delta(\mathcal{C})| = \sum_{v_i \in \mathcal{C}} \delta_i - |\mathcal{C}|(|\mathcal{C}|-1) \leq f(|\mathcal{C}|) \leq f(k_0), \tag{3}$$

where Lemma 2 and Lemma 1 were considered in the last two inequalities. Since the inequalities hold for every clique, in particular we get that $OPT \leq f(k_0)$. ■

Theorem 3 (*Bounds for* c_{min}).
If $\{k_0, k_1\} = \operatorname{argmax} f(k)$ with $k_0 \leq k_1$, then the following inequalities hold for any clique $\mathcal{C}$:

$$\max\{k \leq k_0 : f(k) \leq |\delta(\mathcal{C})|\} \leq c_{min} \leq \min\{k \geq k_1 : f(k) \leq |\delta(\mathcal{C})|\}. \tag{4}$$

Proof. Let $\mathcal{C}'$ be a clique such that $f(|\mathcal{C}'|) \leq |\delta(\mathcal{C})|$ and $|\mathcal{C}'| \leq k_0$. Since $|\delta(\mathcal{C})| \leq OPT \leq f(c_{min})$ and f is strictly increasing in $[1, k_0]$, then $|\mathcal{C}'| \leq c_{min}$. Taking maximum on $|\mathcal{C}'| \leq k_0$, we obtain the first inequality.

The reasoning for the second inequality is analogous. ■

Corollary 1. *If k_0 and k_1 are as in previous Theorem, then the the following inequalities hold:*

$$\max\{k \leq k_0 : f(k) \leq \delta_n\} \leq c_{min} \leq \min\{k \geq k_1 : f(k) \leq \delta_n\} \tag{5}$$

Proof. Apply Theorem 3 with the clique $\mathcal{C} = \{v_n\}$. ■

3 Methodology

GRASP, VND and Tabu Search are well known metaheuristics that have been successfully used to solve many hard combinatorial optimization problems. GRASP is an iterative multi-start process which operates in two phases [7]. In the Construction Phase a feasible solution is built whose neighborhood is then explored in the Local Search Phase [7]. The second phase is usually enriched by means of different variable neighborhood structures. For instance, VND (Variable Neighborhood Descent) explores several neighborhood structures in a deterministic order. Its success is based on the simple fact that different neighborhood structures do not usually have the same local minimum. Thus local optima can be escaped by applying some deterministic rule for altering the neighborhoods [3]. Tabu Search is a strategy to prevent local search algorithms getting trapped in previously visited solutions. It accepts non-improving moves and uses a penalization mechanism called Tabu List [4, 2]. The reader is invited to consult the comprehensive Handbook of Metaheuristic for further information [8].

Here, we develop a GRASP/VND methodology enriched with Tabu Search in order to avoid getting trapped in previous visited solutions. In the following, the Pseudo-code of our Hybrid Metaheuristic (HM) for the max cut-clique is presented (see Algorithm 1). It follows the traditional two-phase GRASP template enriched with a VND (Lines 4-5).

A Tabu Search strategy is included in order to enhance feasible solutions. The tabu list $\mathcal{T}$ stores tabu nodes (Line 6), discarding previous solutions. Essentially, the most frequent nodes involved in all solutions after the second phase (VND) are not considered for further solutions during θ iterations, whenever we reach θ^{max} consecutive iterations without improvement. Most frequent nodes are selected if they appear more than ϕ times since the last tabu list refresh. The real numbers ϕ and θ are uniformly chosen at random in the interval $[1, \theta^{max}]$, being θ^{max} a parameter of the algorithm. The specific GRASP phases for the MCC are described in detail in the following subsections.

Algorithm 1 HM Pseudo-code

Input: α, θ^{max}, *maxIter*, $\mathcal{G}$
Output: $\mathcal{C}^*$

1: $\mathcal{C}^* \leftarrow \emptyset$
2: $\mathcal{T} \leftarrow \emptyset$
3: **for** *iter = 1* **to** *maxIter* **do**
4: $\quad \mathcal{C} \leftarrow$ Clique$(\alpha, \mathcal{T}, \mathcal{G})$
5: $\quad \mathcal{C} \leftarrow$ VND$(\mathcal{C}, \mathcal{T}, \mathcal{G})$
6: $\quad \mathcal{T} \leftarrow$ Update$(\mathcal{T}, \theta^{max}, \mathcal{C})$ ▷ Tabu List
7: $\quad$ **if** $|E'(\mathcal{C})| > |E'(\mathcal{C}^*)|$ **then**
8: $\quad\quad \mathcal{C}^* \leftarrow \mathcal{C}$
9: **return** $\mathcal{C}^*$

3.1 Construction Phase - *Clique*

The construction phase of the proposed algorithm is depicted in Algorithm 2. Let us denote by $\mathcal{C}$ the clique under construction, $\delta(U)$ and $\Delta(U)$ the minimum and maximum degree of the node-set U. The clique $\mathcal{C}$ is initially empty (Line 1), and a multi-start process is considered (Line 2). A Restricted Candidate List (RCL) is defined in Line 3. Observe that the RCL includes nodes with the highest degree, and α trades greediness for randomization. During the *While* loop of Lines 4-11, a singleton $\{i\}$ is uniformly picked from the RCL (Line 5), and the maximum clique $\mathcal{C}'$ is built using the nodes from the set $\mathcal{C} \cup \{i\}$, specifically, $[\mathcal{C} \cap N(i)] \cup \{i\}$, being $N(i)$ the neighbor-set of node i (see Line 6). The best solution is updated if necessary (Lines 7-8). Observe that the process is finished only if we meet $MAX\_ATTEMPTS$ without improvement (Lines 9-11). The reader can appreciate that the output $\mathcal{C}$ is the best feasible clique during the whole process (Line 12).

Algorithm 2 CLIQUE

Input: α, $\mathcal{T}$, $\mathcal{G}$
Output: $\mathcal{C}$

1: $\mathcal{C} \leftarrow \emptyset$
2: $improving = MAX\_ATTEMPTS$
3: $RCL \leftarrow \{v \in V - \mathcal{C} : |E'(v)| \geq \Delta(V - \mathcal{C}) - \alpha(\Delta(V - \mathcal{C}) - \delta(V - \mathcal{C}))\}$
4: **while** $improving > 0$ **do**
5: $\quad i \leftarrow selectRandom(RCL)$
6: $\quad \mathcal{C}' \leftarrow [\mathcal{C} \cap N(i)] \cup \{i\}$
7: $\quad$ **if** $|E'(\mathcal{C}')| > |E'(\mathcal{C})|$ **then**
8: $\quad\quad \mathcal{C} \leftarrow \mathcal{C}'$
9: $\quad\quad improving \leftarrow MAX\_ATTEMPTS$
10: $\quad$ **else**
11: $\quad\quad improving \leftarrow improving - 1$
12: **return** $\mathcal{C}$

3.2 Local Search Phase - *VND*

The goal is to combine a rich diversity of neighborhoods in order to obtain an output that is locally optimum solution for every feasible neighborhood. Five neighborhood structures are considered to build a VND [3]. **Add**, **Swap**, and **Aspiration** are taken from a previous ILS [6]. However, our VND is enriched with 2 additional neighborhood structures, named **Remove** and **Cone**. The following neighborhood take effect whenever the resulting cut-clique is increased:

– **Remove**: a singleton $\{i\}$ is removed from a clique $\mathcal{C}$.
– **Add**: a singleton $\{i\}$ is added from a clique $\mathcal{C}$.
– **Swap**: if we find $j \notin \mathcal{C}$ such that $\mathcal{C} - \{i\} \subseteq N(j)$, we can include j in the clique and delete i (swap i and j).
– **Cone**: generalization of Swap for multiple nodes. The clique $\mathcal{C}$ is replaced by $\mathcal{C} \cup \{i\} - \mathcal{A}$, being $\mathcal{A}$ the nodes from $\mathcal{C}$ that are non-adjacent to i.
– **Aspiration**: this movement offers the opportunity of nodes belonging to the Tabu List to be added.

Observe that the dynamic Tabu list works during the potential additions during **Add**, **Swap** and **Cone**. On the other hand, **Aspiration** provides diversification with an *opportunistic unchoking* process: it picks nodes from the Tabu List instead. For the remaining four local searches, there is an efficient way to determine whether there is an improvement with respect to some neighbor-set. Specifically, the Test Lemmas 3 to 6 are useful to determine the improvements for **Remove**, **Add**, **Swap** and **Cone** movements, respectively. We call Aspiration Test to Lemma 4 but applied in a different domain (specifically, the candidate nodes must belong to the Tabu List).

Lemma 3 (Remove). $|\delta(\mathcal{C} - \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| < 2(|\mathcal{C}| - 1)$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \{i\})| &= |\delta(\mathcal{C})| + |\mathcal{C}| - 1 - (|\delta(i)| - (|\mathcal{C}| - 1)) \\ &= |\delta(\mathcal{C})| + |\mathcal{C}| - 1 - |\delta(i)| + |\mathcal{C}| - 1 \\ &= |\delta(\mathcal{C})| + 2(|\mathcal{C}| - 1) - |\delta(i)| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $2(|\mathcal{C}| - 1) - |\delta(i)| > 0$. ■

Lemma 4 (Add). $|\delta(\mathcal{C} \cup \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > 2|\mathcal{C}|$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} \cup \{i\})| &= |\delta(\mathcal{C})| - |\mathcal{C}| + |\delta(i)| - |\mathcal{C}| \\ &= |\delta(\mathcal{C})| + |\delta(i)| - 2|\mathcal{C}| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > 2|\mathcal{C}|$. ■

Lemma 5 (Swap). $|\delta(\mathcal{C} - \{j\} \cup \{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > |\delta(j)|$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \{j\} \cup \{i\})| &= |\delta(\mathcal{C})| - |\delta(j)| + 2(|\mathcal{C}| - 1) + |\delta(i)| - 2(|\mathcal{C}| - 1) \\ &= |\delta(\mathcal{C})| - |\delta(j) + |\delta(i)| \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > |\delta(j)|$. ■

Lemma 6 (Cone). $|\delta(\mathcal{C}-\mathcal{A}\cup\{i\})| > |\delta(\mathcal{C})|$ *iff* $|\delta(i)| > |\delta(\mathcal{A})|-2|\mathcal{C}-\mathcal{A}|(|\mathcal{A}|-1)$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} - \mathcal{A} \cup \{i\})| &= |\delta(\mathcal{C})| + |\mathcal{A}||\mathcal{C} - \mathcal{A}| - (|\delta(\mathcal{A})| - |\mathcal{A}||\mathcal{C} - \mathcal{A}|) - 2|\mathcal{C} - \mathcal{A}| + |\delta(i)| \\ &= |\delta(\mathcal{C})| + 2|\mathcal{A}||\mathcal{C} - \mathcal{A}| - |\delta(\mathcal{A})| - 2|\mathcal{C} - \mathcal{A}| + |\delta(i)| \\ &= |\delta(\mathcal{C})| + 2|\mathcal{C} - \mathcal{A}|(|\mathcal{A}| - 1) - |\delta(\mathcal{A})| + |\delta(i)||\delta(\mathcal{C} - \mathcal{A} \cup \{i\})| \\ &> |\delta(\mathcal{C})| \end{aligned}$$

where the last inequality holds iff $|\delta(i)| > |\delta(\mathcal{A})| - 2|\mathcal{C} - \mathcal{A}|(|\mathcal{A}| - 1)$. ■

The Flow Diagram of our VND is presented in Figure 1. The ordered sequence of local searches are **Remove**, **Add**, **Swap**, **Cone** and **Aspiration** moves. Once an improvement is obtained, the process restarts from the beginning. Observe that, in the output, a locally optimum solution under all neighborhood structures is met.

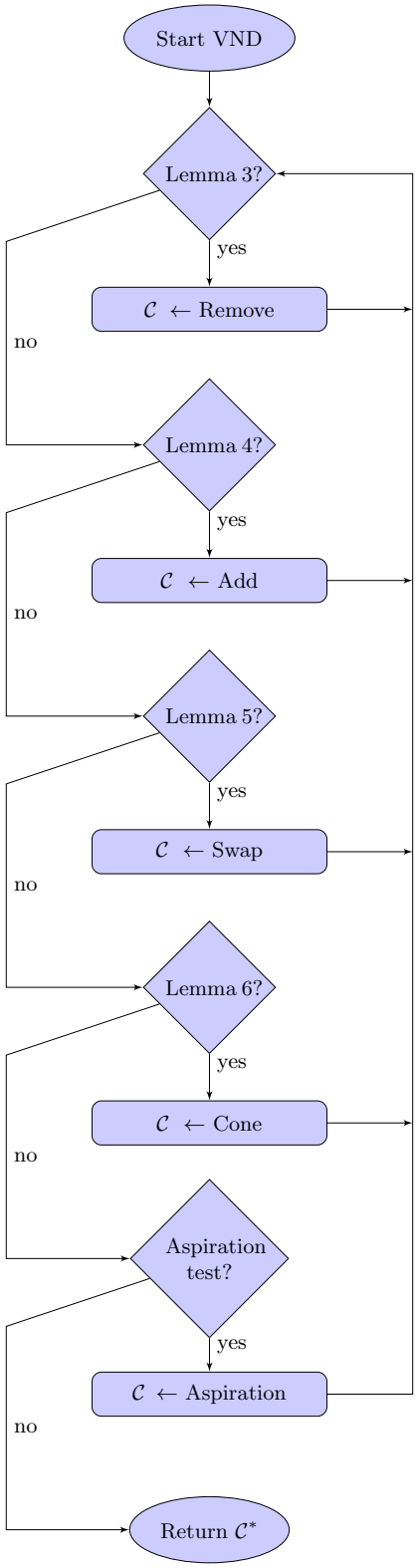


Fig. 1. Flow Diagram for the Local Search Phase - VND.

4 Exact Method for the MCC

In this section, we present an exact method based on a mathematical formulation. Due to combinatorial nature, we addressed it by integer programming, using the following decision variables:

$$w_i = \begin{cases} 1 & \text{if node } i \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}, \forall i \in V$$

$$w_{(i,j)} = \begin{cases} 1 & \text{if edge } (i,j) \in E(\mathcal{C}) \\ 0 & \text{otherwise} \end{cases}, \forall (i,j) \in E$$

An integer programming model is presented below. Constraint (1) and (2) state that both nodes i, j belong to the clique $\mathcal{C}$ if and only if $(i,j) \in E(\mathcal{C})$. Recall that Theorem 3 provides a feasible interval for the size of the clique, c_{min}. Constraints (3) and (4) determine lower and upper bounds Lb and Ub for the size of the clique, found combining Theorem 3 and the best output of our GRASP/VND heuristic. Constraints (5) and (6) just state that w_i and $w_{(i,j)}$ are binary variables. The goal is maximize the cut-clique, which is precisely the difference between the sum-degree minus twice the number of internal links.

$$\begin{array}{lll}
\max \sum_{i \in V} d_i \times w_i - 2 \times \sum_{(i,j) \in E} w_{(i,j)} & & \\
\text{s.a. } 2w_{(i,j)} \leq w_i + w_j & \forall (i,j) \in E & (1) \\
w_i + w_j - 1 \leq w_{(i,j)} & \forall i,j \in V & (2) \\
 & & \\
\sum_{i \in V} w_i \geq Lb & & (3) \\
\sum_{i \in V} w_i \leq Ub & & (4) \\
 & & \\
w_{(i,j)} \in \{0,1\} & \forall (i,j) \in E & (5) \\
w_i \in \{0,1\} & \forall i \in V & (6)
\end{array}$$

5 Computational Results

In order to test the performance of the algorithm we carried out a fair comparison with respect to the ILP model implemented using IBM CPLEX 12.8. Both algorithms are executed on a Home-PC (Intel Core i7, 2.4 GHz, 8GB RAM). The graphs under study were obtained from the SteinLib [1] and DIMACS.

Table 1 reports the performance of both algorithms for each instance. All HM algorithm instances were tested using a single run with one-hundred iterations and $\alpha = \frac{1}{2}$, $MAX\_ATTEMPTS = \lfloor \frac{|V|}{10} \rfloor$, $\theta^{max} = 4$. Lower and upper bounds Lb and Ub were obtained for each topology under study using Corollary 1.

The values remarked using bold letters from column $|\delta(\mathcal{C})|$ indicate that the best solution was reached according to the output from the ILP solver.

Following the terminology, $|\delta(\mathcal{C})|$, $|\mathcal{C}|$ and $Time$ represent maximum cut-clique size found, best solution, and the CPU time for the best solution found. Lb, Ub columns are reported for the ILP solver which represents the lower an upper bound for the ILP model. Under ILP, $Time$ give the time to reach the optimum value or the best lower bound to the optimum when the optimum is not attained within the given time limit (10800 seconds).

The reader can appreciate from Table 1 that our GRASP/VND algorithm meets the best solution in all cases. The globally optimum for all the instances under study is formally proved using the ILP formulation. Furthermore, our GRASP/VND approach presents consistently smaller CPU times for graphs with large size.

Table 2 shows the performance of the VND algorithm. The activity of every single local search is studied. Swap and Add movements show to be more effective, while Remove and Cone take effect few times. Aspiration has no effect, but it works for dense graphs.

In order to understand the global effectiveness of our VND scheme, a *mid-point test* is performed. The columns Remove, Add, Swap, Cone and Aspiration show the percentage of each kind of movement applied over one-hundred executions of the VND local search phase. The column #moves states the amount of movement applied during these iterations. The column entitled *mp* displays the average gap in percentage between the best solution found in each local search phase with respect to the feasible solution obtained from the construction phase over one-hundred iterations. The reader can appreciate that the VND effect is notorious, since the cut-clique is roughly half the optimum in most cases using only the Construction Phase.

It is worth to remark that we further studied the performance of our GRASP/VND methodology versus a state-of-the-art ILS heuristic for the MCC, detailed in [6]. We could find optimality in all the reported instances which achieved optimality, and we found the best feasible solutions so far in the remaining cases, with identical results offered in [6].

[1] The dataset can be found in the URL http://steinlib.zib.de/steinlib.php

Instances			GRASP/VND			ILP				
name	n	Density	$\lvert E'(\mathcal{C})\rvert$	$\lvert\mathcal{C}\rvert$	Time (s)	Lb	Ub	$\lvert E'(\mathcal{C})\rvert$	$\lvert\mathcal{C}\rvert$	Time (s)
i080-001	80	0.039	**13**	2	0.7120	2	8	**13**	2	0.66
i080-002	80	0.039	**13**	2	1.4779	2	9	**13**	2	0.62
i080-011	80	0.11	**38**	4	3.0396	3	16	**38**	4	0.94
i080-044	80	0.2	**80**	5	1.1495	4	26	**80**	5	0.71
i080-045	80	0.2	**74**	4	0.5748	4	25	**74**	4	0.90
i080-111	80	0.11	**35**	4	0.4141	3	15	**35**	4	0.88
i080-112	80	0.11	**39**	3	0.3191	3	19	**39**	3	0.64
i080-131	80	0.05	**16**	2	1.1908	2	10	**16**	2	0.73
i080-132	80	0.05	**15**	3	0.5146	2	9	**15**	3	0.69
i080-142	80	0.2	**74**	4	1.0306	4	25	**74**	4	0.70
i080-143	80	0.2	**80**	4	0.8546	4	26	**80**	4	0.85
i160-001	160	0.019	**15**	2	0.6351	2	10	**15**	2	2.84
i160-002	160	0.019	**14**	2	2.8054	2	9	**14**	2	2.44
i160-011	160	0.064	**44**	3	4.6177	3	20	**44**	3	24.53
i160-044	160	0.2	**180**	5	5.7298	5	47	**180**	5	11.23
i160-045	160	0.2	**173**	5	3.8451	5	42	**173**	5	14.92
i160-111	160	0.064	**50**	4	5.9956	4	18	**50**	4	5.59
i160-112	160	0.064	**46**	4	0.47	3	18	**46**	4	5.71
i160-131	160	0.025	**19**	3	2.5908	3	10	**19**	3	2.80
i160-132	160	0.025	**22**	3	2.3052	3	12	**22**	3	2.95
i160-142	160	0.2	**183**	5	3.7192	5	45	**183**	5	10.63
i160-143	160	0.2	**170**	5	05.097	5	44	**170**	5	10.57
mc11	400	0.0095	**6**	2	0.7292	2	2	**6**	2	6.43
c-fat200-1	200	0.077	**81**	9	0.1860	9	17	**81**	9	29.57
c-fat200-2	200	0.163	**306**	17	0.8388	17	34	**306**	17	88.79
c-fat200-5	200	0.426	**1892**	43	13.0593	43	86	**1892**	43	1717.39
c-fat500-1	500	0.036	**110**	10	4.77459	10	20	**110**	10	1198.31
c-fat500-2	500	0.073	**380**	19	14.1875	19	38	**380**	19	1822.08
c-fat500-5	500	0.186	2304	48	121.32	48	95	2304	48	10800
c-fat500-10	500	0.374	8930	94	33.298	94	188	8930	94	10800

Table 1. HM versus ILP for the MCC.

Instances			GRASP/VND						
name	n	Density	Remove (%)	Add (%)	Swap (%)	Cone (%)	Aspiration (%)	#moves	mp (%)
i080-001	80	0.039	0	38	57	5	0	154	30.949
i080-002	80	0.039	3	32	59	7	0	147	27.571
i080-011	80	0.11	1	48	48	3	0	158	18.647
i080-044	80	0.2	0	54	46	0	0	239	17.390
i080-045	80	0.2	0	56	44	0	0	217	20.585
i080-111	80	0.11	0	59	39	3	0	176	22.572
i080-112	80	0.11	0	48	37	5	0	244	16.261
i080-131	80	0.05	1	30	60	9	0	151	25.983
i080-132	80	0.05	0	32	68	0	0	136	23.402
i080-142	80	0.2	0	50	50	0	0	202	19.762
i080-143	80	0.2	0	52	48	0	0	253	18.467
i160-001	160	0.019	0	22	73	5	0	143	29.452
i160-002	160	0.019	0	20	80	0	0	135	26.427
i160-011	160	0.064	0	57	40	3	0	186	21.803
i160-044	160	0.2	0	60	40	0	0	251	18.079
i160-045	160	0.2	0	53	47	0	0	211	16.358
i160-111	160	0.064	0	61	38	1	0	181	23.816
i160-112	160	0.064	0	55	45	0	0	154	23.940
i160-131	160	0.025	0	39	58	2	0	168	23.710
i160-132	160	0.025	0	42	54	3	0	179	26.588
i160-142	160	0.2	0	57	42	0	0	250	18.146
i160-143	160	0.2	0	59	41	0	0	196	17.664
mc11	400	0.0095	0	100	0	0	0	46	50.000
c-fat200-1	200	0.077	0	100	0	0	0	376	13.159
c-fat200-2	200	0.163	0	100	0	0	0	475	6.338
c-fat200-5	200	0.426	0	99	1	0	0	138	0.505
c-fat500-1	500	0.036	0	100	0	0	0	391	12.965
c-fat500-2	500	0.073	0	100	0	0	0	278	3.736
c-fat500-5	500	0.186	2	98	0	0	0	132	3.252
c-fat500-10	500	0.374	0	100	0	0	0	11	0.132

Table 2. Performance of the Local Search Phase.

6 Conclusions and Trends for Future Work

Several business models can be represented by Market Basket Analysis (MBA). A relevant marketing approach is to find a subset of items that are strongly correlated with the others. This intuition is formalized by means of a combinatorial optimization problem, called Max Cut-Clique (MCC).

In this paper, the $\mathcal{NP}$-Completeness of MCC is established. This fact promotes the development of heuristics and bounds. As a consequence, we offered bounds for both the globally optimum solution and the size of the minimum cardinality clique with maximum cut. Then, a GRASP/VND methodology enriched with Tabu Search is developed to address the MCC. A fair comparison with an exact ILP formulation confirms the optimality of our approach for hundreds of nodes. Furthermore, the computational effort is reduced for the heuristic under large-sized graphs. The movements Swap and Add have the largest activity for the instances under study. The experiments shows that our GRASP/VND heuristic is competitive with state-of-the-art solutions for the MCC. Further analysis should be done to determine the best order for the VND in terms of computational efficiency.

As future work we would like to implement our solution into a real-life product-placement scenario. In a first stage, we need historical information to determine the links between pairs of items. The physical location of the items must be determined using a complementary geometrical problem with constraints. The solution could consider multi-constrained clustering in order to include categories for the items, or other Machine Learning techniques to determine profiles for the customers, according to the product under study. After the real implementation, the feedback of sales in a period is a valuable metric of success.

7 Acknowledgements

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos*.

References

1. Herman Aguinis, Lura E. Forcum, and Harry Joo. Using market basket analysis in management research. *Journal of Management*, 39(7):1799–1824, 2013.
2. A. Amuthan and K. Deepa Thilak. Survey on tabu search meta-heuristic optimization. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pages 1539–1543, Oct 2016.
3. Abraham Duarte, Nenad Mladenović, Jesús Sánchez-Oro, and Raca Todosijević. *Variable Neighborhood Descent*, pages 1–27. Springer International Publishing, Cham, 2016.
4. Fred Glover and Manuel Laguna. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA, 1997.

5. Pedro Martins. Cliques with maximum/minimum edge neighborhood and neighborhood density. *Computers & Operations Research*, 39(3):594 – 608, 2012.
6. Pedro Martins, Antonio Ladrón, and Helena Ramalhinho. Maximum cut-clique problem: ILS heuristics and a data analysis application. *International Transactions in Operational Research*, 22(5):775–809.
7. Mauricio G.C. Resende and Celso C. Ribeiro. *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*. Computational Science and Engineering. Springer-Verlag New York, 2016.
8. Said Salhi. Handbook of metaheuristics (2nd edition). *Journal of the Operational Research Society*, 65(2):320–320, Feb 2014.

Chapter 4

Complexity and Heuristics for the Weighted Max Cut-Clique Problem

In this chapter, a generalization of the *MCC* is established considering the weights associated with each link. In this context we are interested in finding the clique $\mathcal{C} \subset V$ such that the weighted-sum associated to each link shared between $\mathcal{C}$ and $V \setminus \mathcal{C}$ is maximized. The weighted version of *MCC* is known as *MEWNC* (Maximum Edge-Weight Neighborhood Clique).

INTERNATIONAL TRANSACTIONS IN OPERATIONAL RESEARCH

Intl. Trans. in Op. Res. XX (2018) 1–21
DOI: xx.xxxx/itor.xxxxx

Complexity and Heuristics for the Weighted Max Cut-Clique Problem

Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero, Luis Stábile

Instituto de Matemática y Estadística, IMERL
Facultad de Ingeniería - Universidad de la República
Montevideo Uruguay
E-mail: (mbourel,canale,frobledo,promero,lstabile)@fing.edu.uy

Received DD MMMM YYYY; received in revised form DD MMMM YYYY; accepted DD MMMM YYYY

Abstract

In Market Basket Analysis (MBA), the goal is to understand the human behavior in order to maximize sales. An evident behavior is to buy correlated items. As a consequence, the determination of a set of items with a large correlation with others is a valuable tool for MBA.

In this work we address a combinatorial optimization problem with valuable applications to MBA, specially in marketing and product-placement. For any given graph $G = (V, E)$ (where the nodes are items and links represent correlation), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized. This problem is known in the literature as Max Cut-Clique (*MCC*). We can generalize this problem considering the weights associated with each link. In this context we are interested in finding the clique $\mathcal{C} \subseteq V$ such that the weighted-sum associated to each link shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized. The weighted version of *MCC* is known as *MEWNC* (Maximum Edge-Weight Neighborhood Clique).

The contributions of this paper are four-fold. First, the computational complexity of both *MCC* and *MEWNC*, are established. Specifically, we prove that the *MCC* and *MEWNC* belong to the class of $\mathcal{NP}$-Complete problems. Second, an exact Integer Linear Programming (ILP) formulation for the *MEWNC* is offered. Third, a full GRASP/VND methodology enriched with a Tabu Search is here developed, where the main ingredients are novel local searches and a Restricted Candidate List that trades greediness for randomization in a multi-start fashion. A dynamic Tabu list considers a bounding technique based on the previous analysis.

Finally, a fair comparison between our hybrid algorithm and a globally optimal solution using the ILP formulation confirms that a globally optimal solution is found by our heuristic for graphs with hundreds of nodes, but it is more efficiently than the exact solution in terms of time and memory requirements.

Keywords: Market Basket Analysis, Combinatorial Optimization Problem, Max Cut-Clique, ILP, GRASP, Tabu Search.

*Luis Stábile (lstabile@fing.edu.uy).

1. Introduction

There is a serious disconnection between the knowledge that academics are producing and the knowledge that practitioners are consuming (Cascio and Aguinis (2008)). A bridge between the science-practice division can be found in Market Basket Analysis (MBA), sometimes known as *affinity analysis* (Aguinis et al. (2013)). In synthesis, MBA is a Data Mining technique (Agrawal et al. (1993); Tan et al. (2005)) originated in the field of marketing. It has recent applications to other fields, such as bioinformatics (Bader and Hogue (2003); Brohée and van Helden (2006)), WWW networks (Henzinger and Lawrence (2004)), criminal networks (Bruinsma and Bernasco (2004)) and financial networks (Hüffner et al. (2008)). The goal of MBA is to identify non-obvious or counterintuitive relationships between groups of products, items, or categories.

The information obtained from MBA impacts in the business model and operations. It commonly suggests a new approach to product placement solutions, optimal product-line offering, personalized marketing campaigns and product promotions. The analysis is commonly supported by Machine Learning (pattern matching, clustering, feature extraction, statistics), Optimization and Logical rules for association.

This work is focused on a specific combinatorial optimization methodology to assist product placement; however, related applications could be found. The problems under study are called Max Cut-Clique (*MCC*) and Weighted Max Cut-Clique respectively. The first one was introduced by Martins (2012) and the second one was introduced by Martins and Gouveia (2015). For any given graph $G = (V, E)$ (where the nodes are items and links represent correlation with their respective weights), we want to find the clique $\mathcal{C} \subseteq V$ such that the number of links (or the weighted-sum in the second case) shared between $\mathcal{C}$ and $V - \mathcal{C}$ is maximized.

In (Martins (2012)), the author states that the *MCC* is presumably hard, since related problems such as *MAX-CUT* and *MAX-CLIQUE* are both $\mathcal{NP}$-Complete. To the best of our knowledge, there is no formal proof available for the hardness of the *MCC* in the published scientific literature. Nevertheless, the *MCC* is systematically addressed by the scientific community with metaheuristics and exact solvers that run in exponential time.

A recent work in the field develops an Iterated Local Search for the *MCC* (Martins et al. (2015)). As far as we know, this work belongs to the state-of-the-art techniques for the *MCC*. The authors find optimal solutions for most instances under study, and suggest a rich number of applications.

The contributions of this paper can be summarized in the following items:

1. The $\mathcal{NP}$-Completeness of both *MCC* and *MEWNC* are established (Subsection 2.1).
2. Bounds for both a globally optimal solution and the clique size are produced (Subsection 2.2).
3. A hybrid GRASP/VND heuristic enriched with Tabu Search is developed to address the *MCC* and *MEWNC* (Section 3).
4. An exact Integer Linear Programming (ILP) formulation for the *MEWNC* is proposed (Section 4).
5. Our GRASP/VND is not only faster than the exact solution, but also achieves optimality under instances with hundreds of nodes (Section 5).
6. A fair comparison with a state-of-the-art heuristic is also presented (Section 6).

7. A discussion of potential applications for product-placement is included (Section 7).

2. Analysis and Complexity

In this section, the computational complexity for both the *MCC* and *MEWNC* are established. We formally prove that the *MEWNC* accepts a reduction from PARTITION, while the *MCC* accepts a reduction from MAX-CLIQUE (Subsection 2.1). Then, we find bounds for the *MCC* using elementary graph theory (Subsection 2.2).

It is worth to remark that the hardness promotes the development of heuristics, and these bounds will enrich our GRASP/VND heuristic with a dynamic Tabu List.

2.1. Complexity

The cornerstone in computational complexity is Cook Theorem (Cook, 1971) and Karp reducibility among combinatorial problems (Karp, 1972). Stephen Cook formally proved that the joint satisfiability of an input set of clauses in disjunctive form is an $\mathcal{NP}$-Complete decision problem. Furthermore, he provided a systematic procedure to prove that a certain problem is $\mathcal{NP}$-Complete. Specifically, it suffices to prove that the decision problem belongs to set $\mathcal{NP}$, and that it is at least as hard as an $\mathcal{NP}$-Complete problem. Richard Karp followed this hint, and presented the first 21 combinatorial problems that belong to this class. In particular, *PARTITION* belongs to this list. The reader is invited to consult the classical book in Complexity Theory (Garey and Johnson, 1979), which has a larger list of $\mathcal{NP}$-Complete problems, a rich number of bibliographic references and examples.

Consider the two following auxiliary decision problems:

Definition 1 (*PARTITION*). GIVEN: a set of natural numbers $A = \{a_1, \ldots, a_n\}$.
QUESTION: is there a subset $S \subset A$ such that $\sum_{a_i \in S} a_i = \sum_{a_j \notin S} a_j$?

Definition 2 (*MAX-CLIQUE*). GIVEN: a graph $G = (V, E)$ and a number $K \in \mathbb{R}$.
QUESTION: is there a clique $\mathcal{C} \subseteq V$ such that $|\mathcal{C}| \geq K$?

Here, we formally prove that the decision versions for both *MCC* and *MEWNC* are $\mathcal{NP}$-Complete. Let us denote for convenience $\delta(\mathcal{C})$ as the cut produced by a node-set $\mathcal{C}$.

Definition 3 (*MCC*). GIVEN: a graph $G = (V, E)$ and a real number K.
QUESTION: is there a clique $\mathcal{C} \subseteq G$ such that $|\delta(\mathcal{C})| \geq K$?

Definition 4 (*MEWNC*). GIVEN: a weighted graph $G = (V, E, w)$ with $w : E \to \mathbb{N}$ and a number $K \in \mathbb{R}$.
QUESTION: is there a clique $\mathcal{C} \subseteq G$ such that $\sum_{i \in \mathcal{C}, j \notin \mathcal{C}, e=(i,j), e \in E} w(e) \geq K$?

Observe that the *MCC* is precisely the *MEWNC* when $w(e) = 1$ for all $e \in E$. It is clear that, by inclusion, the hardness of the *MCC* implies the hardness of the *MEWNC*. Since we chronologically discovered the hardness of the latter first, here we present both proofs. Basically, *MEWNC* accepts a reduction from PARTITION such as in the foundational work from Karp (1972), while *MCC* accepts a reduction from MAX-CLIQUE:

Theorem 1. The *MEWNC* belongs to the class of $\mathcal{NP}$-Complete problems.

Proof. Given a natural subset $\mathcal{A} = \{a_1, \ldots, a_n\}$, let us denote $s = \sum_{i=1}^{n} a_i$. We consider the instance (G, w, K) for the *MEWNC* with $G = K_n$, $e = (i, j)$, $w(e) = a_i \times a_j$ for every link e and $K = s^2/4$. The clique $\mathcal{C}$ that achieves the maximum cut accepts the following factorization:

$$\max_{\mathcal{C}} \sum_{i \in \mathcal{C}, j \notin \mathcal{C}} a_i \times a_j = \max \sum_{i \in \mathcal{C}} a_i \sum_{j \notin \mathcal{C}} a_j = A \times B,$$

where $A + B = s$ is constant. But the product $A \times B$ is maximized exactly when $A = B = s/2$, this is, when $PARTITION$ is solved. In that case, the answer for the *MEWNC* is positive, since the max cut-clique has capacity $A \times B = s^2/4 \geq K$.

We proved that the answer for *MEWNC* for the instance (G, w, K) is positive if and only if $PARTITION$ has a positive answer. Therefore, *MEWNC* is at least as hard as $PARTITION$. Both problems clearly belong to the set of $\mathcal{NP}$ decision problems. Then, *MEWNC* is $\mathcal{NP}$-Complete, as desired. □

Theorem 2. The *MCC* belongs to the class of $\mathcal{NP}$-Complete problems.

Proof. We prove that the *MCC* is at least as hard as *MAX-CLIQUE*. Consider a simple graph $G = (V, E)$ with order $n = |V|$ and size $m = |E|$. Let us connect m leaf-nodes hanging to every single node $v \in V$ (observe that there are $m \times n$ such nodes). The resulting graph is called H. If we find a polynomial-time algorithm for *MCC*, then we can produce the max cut-clique in H. But observe that the max cut-clique $\mathcal{C}$ in H must belong to G. If $\mathcal{C}$ has cardinality c, then the cut-clique has precisely $c \times m$ hanging nodes. By construction, the cut-clique must maximize the number of hanging nodes, since the whole size $|E| = m$ is added to the cut by a single addition of a node in the clique. As a consequence, c must be the *MAX-CLIQUE*. We proved that the *MCC* is at least as hard as *MAX-CLIQUE*, as desired. Since *MCC* belongs to the set of $\mathcal{NP}$ decision problems, it belongs to the $\mathcal{NP}$-Complete class. □

2.2. Bounds for MCC

Observe that a global optimum for the *MCC* could be attained by more than one clique. Let us denote by $\mathcal{C}_{min}$ the minimum cardinality clique such that $|\delta(\mathcal{C}_{min})| = OPT$, the optimal value for the *MCC*, and $c_{min} = |\mathcal{C}_{min}|$.

Definition 5. A finite sequence $\{a_i\}_{i=0}^{n}$ is *strictly unimodal* if there exists some index $k_0 \in \{0, \ldots, n\}$ such that $a_0 < a_1 < \cdots < a_{k_0}$ and $a_{k_0} \geq a_{k_0+1} > a_{k_0+2} > \cdots > a_n$.

Lemma 1. Consider a connected graph G with degree-sequence $(\delta_1, \ldots, \delta_n)$, where for convenience we consider $\delta_1 \leq \delta_2 \leq \cdots \leq \delta_n$. Then, the following finite sequence $\{f(k)\}_{k=0}^{n}$ is strictly unimodal, where

$$f(k) = -k(k-1) + \sum_{i=1}^{k} \delta_{n-i+1}, \ \forall k \in \{0, 1, \ldots, n\}. \tag{1}$$

Proof. The difference between consecutive terms, $\Delta_k = f(k) - f(k-1)$, is:

$$\Delta_k = -k(k-1) + \sum_{i=1}^{k} \delta_{n-i+1} + (k-1)(k-2) - \sum_{i=1}^{k-1} \delta_{n-i+1} = -2(k-1) + \delta_{n-k+1}.$$

Since $-2(k-1)$ and δ_{n-k+1} are monotonically decreasing sequences, being the former strictly decreasing, $\{\Delta_k\}_{k\geq 0}$ must be strictly decreasing as well. Furthermore, since $\Delta_1 = f(1) - f(0) = \delta_n > 0$ and $\Delta_n = -2(n-1) + \delta_1 < 0$, there exists some index k_0 such that: $f(0) < f(1) < \cdots < f(k_0)$ and $f(k_0) \geq f(k_0+1) > f(k_0+2) > \cdots > f(n)$, as desired. □

Lemma 2. The following inequalities hold for any clique $\mathcal{C}$:

$$-|\mathcal{C}|(|\mathcal{C}|-1) - \sum_{i=1}^{|\mathcal{C}|} \delta_i \leq |\delta(\mathcal{C})| \leq f(|\mathcal{C}|) \tag{2}$$

Proof. The sum $|\delta(\mathcal{C})| + |\mathcal{C}|(|\mathcal{C}|-1) = \sum_{v_i \in \mathcal{C}} \delta_i$ is greater (smaller) than the sum of the $|\mathcal{C}|$ smallest (greatest) degrees. □

In the following, we will provide an upper-bound for OPT, the optimal value for the *MCC*, and bounds for the size c_{min} of the minimum cardinality clique $\mathcal{C}_{min}$, in terms of the auxiliary sequence $\{f(k)\}_{k=0}^{n}$.

Theorem 3 (*Upper-Bound for MCC*). If OPT denotes the optimal value for the *MCC* and f is maximized at k_0, then $OPT \leq f(k_0)$.

Proof. If we are given an arbitrary clique $\mathcal{C}$, the incident links to some $v \in \mathcal{C}$ either belong to the clique or to the cut. Then:

$$|\delta(\mathcal{C})| = \sum_{v_i \in \mathcal{C}} \delta_i - |\mathcal{C}|(|\mathcal{C}|-1) \leq f(|\mathcal{C}|) \leq f(k_0), \tag{3}$$

where Lemma 2 and Lemma 1 were considered in the last two inequalities. Since the inequalities hold for every clique, in particular we get that $OPT \leq f(k_0)$. □

Theorem 4 (*Bounds for* c_{min}). If $\{k_0, k_1\} = \operatorname{argmax} f(k)$ with $k_0 \leq k_1$, then the following inequalities hold for any clique $\mathcal{C}$:

$$\max\{k \leq k_0 : f(k) \leq |\delta(\mathcal{C})|\} \leq c_{min} \leq \min\{k \geq k_1 : f(k) \leq |\delta(\mathcal{C})|\}. \tag{4}$$

Proof. Let $\mathcal{C}'$ be a clique such that $f(|\mathcal{C}'|) \leq |\delta(\mathcal{C})|$ and $|\mathcal{C}'| \leq k_0$. Since $|\delta(\mathcal{C})| \leq OPT \leq f(c_{min})$ and f is strictly increasing in $[1, k_0]$, then $|\mathcal{C}'| \leq c_{min}$. Taking maximum on $|\mathcal{C}'| \leq k_0$, we obtain the first inequality. The reasoning for the second inequality is analogous. □

Corollary 1. If k_0 and k_1 are as in previous Theorem, then the following inequalities hold:

$$\max\{k \leq k_0 : f(k) \leq \delta_n\} \leq c_{min} \leq \min\{k \geq k_1 : f(k) \leq \delta_n\} \tag{5}$$

Proof. Apply Theorem 4 with the clique $\mathcal{C} = \{v_n\}$. □

2.3. Bounds for MEWNC

Let T_k be the k^{th} triangle number, i.e.,

$$T_k = \frac{(k+1)k}{2},$$

and let us denote by $\mathcal{C}^{\omega}_{min}$ the minimum cardinality clique such that $\omega(\delta(\mathcal{C}^{\omega}_{min})) = OPT^{\omega}$, the optimal value for the *MEWNC*, and $c^{\omega}_{min} = |\mathcal{C}^{\omega}_{min}|$.

Lemma 3. Consider a connected graph G with vertices $v_1, \ldots, v_n$ and edges $e_1, \ldots, e_m$ such that $\delta^{\omega}_1 \leq \ldots \leq \delta^{\omega}_n$ where $\delta^{\omega}_i = \omega(\delta(v_i)))$, and $\omega_1 \leq \omega_2 \leq \cdots \leq \omega_m$, with $\omega_i = \omega(e_i)$.

Then, the following finite sequence $\{g(k)\}_{k=0}^{n_0}$ is either strictly unimodal or increasing, where $n_0 = \max\{k : T_k \leq m\}$ and

$$g(k) = \sum_{i=1}^{k} \delta^{\omega}_{n-i+1} - \sum_{i=1}^{T_{k-1}} \omega_i.$$

Proof. We proceed by computing the difference between consecutive terms of $g(k)$:

$$\Delta g(k) = g(k+1) - g(k) = \delta^{\omega}_{n-k} - s_k \quad \text{with } s_k = \sum_{i=T_{k-1}+1}^{T_k} \omega_i$$

Since δ^{ω}_{n-k} is monotonically decreasing, it suffices to prove that s_k is strictly increasing. Since w_i is an increasing sequence and $T_k - T_{k-1} = k$, the sequence s_k is monotonically increasing, the result holds. □

Theorem 5. The following inequalities hold for any clique $\mathcal{C}$ with cardinality $c = |\mathcal{C}|$:

$$\omega(\delta(\mathcal{C})) \leq g(c). \tag{6}$$

Proof. For any clique $\mathcal{C}$, the incident links to some $v \in \mathcal{C}$ either belong to the clique or to the cut $\delta(\mathcal{C})$, thus

$$\omega(\delta(\mathcal{C})) = -\sum_{x,y \in \mathcal{C}} \omega(xy) + \sum_{v_i \in \mathcal{C}} \delta^{\omega}_i$$

The sum of the T_{c-1} weights $\omega(xy)$ (respectively c weighted degrees δ^{ω}_i) is greater (resp. smaller) than the sum of the T_{c-1} smallest weights $\omega_1, \ldots \omega_{T_{c-1}}$ (respectively c greater weighted degrees $\delta^{\omega}_n, \ldots, \delta^{\omega}_{n-c+1}$). Thus

$$\omega(\delta(\mathcal{C})) \leq -\sum_{i=1}^{T_{c-1}} \omega_i + \sum_{i=1}^{c} \delta^{\omega}_{n-i+1} = g(c).$$

□

In the following, we will provide an upper-bound for OPT^{ω}, the optimal value for the *MEWNC*, and bounds for the size c^{ω}_{min} of the minimum cardinality clique $\mathcal{C}^{\omega}_{min}$, in terms of the auxiliary sequence $\{g(k)\}_{k=0}^{n_0}$.

Corollary 2 (*Upper-Bound for MEWNC*). If OPT^{ω} denotes the optimal value for the *MEWNC* and g is maximized at k_0, then $OPT^{\omega} \leq g(k_0)$. □

Theorem 6 (*Bounds for* c^{ω}_{min}). If $\{k_0, k_1\} = \operatorname{argmax} g(k)$ with $k_0 \leq k_1$, then the following inequalities hold for any clique $\mathcal{C}$:

$$\max\{k \leq k_0 : g(k) \leq \omega(\delta(\mathcal{C}))\} \leq c^{\omega}_{min} \leq \min\{k \geq k_1 : g(k) \leq \omega(\delta(\mathcal{C}))\}. \tag{7}$$

Proof. Let $\mathcal{C}'$ be a clique such that $g(|\mathcal{C}'|) \leq \omega(\delta(\mathcal{C}))$ and $|\mathcal{C}'| \leq k_0$. Since $\omega(\delta(\mathcal{C})) \leq OPT^{\omega} \leq g(c^{\omega}_{min})$ and g is strictly increasing in $[1, k_0]$, then $|\mathcal{C}'| \leq c^{\omega}_{min}$. Taking maximum on $|\mathcal{C}'| \leq k_0$, we obtain the first inequality. The reasoning for the second inequality is analogous. □

3. Methodology

GRASP, VND and Tabu Search are well known metaheuristics that have been successfully used to solve many hard combinatorial optimization problems. GRASP is an iterative multi-start process which operates in two phases (Resende and Ribeiro (2016)). In the Construction Phase a feasible solution is built whose neighborhood is then explored in the Local Search Phase (Resende and Ribeiro (2016)). The second phase is usually enriched by means of different variable neighborhood structures. For instance, VND (Variable Neighborhood Descent) explores several neighborhood structures in a deterministic order. Its success is based on the simple fact that different neighborhood structures do not usually have the same local minimum. Thus, the local optima trap problem may be resolved by deterministic change of neighborhoods (Duarte et al. (2016)). Tabu Search is a strategy to prevent local search algorithms getting trapped in previously visited solutions. It operates accepting non improving moves and uses a penalization mechanism called Tabu List (Glover and Laguna (1997); Amuthan and Thilak (2016)). The reader is invited to consult the comprehensive Handbook of Metaheuristic for further information (Salhi (2014)).

Here, we develop a GRASP/VND methodology enriched with Tabu Search in order to avoid getting trapped in previous visited solutions. In the following, the Pseudo-code of our Hybrid Metaheuristic (HM) for the max cut-clique is presented (see Algorithm 1). It follows the traditional two-phase GRASP template enriched with a VND (Lines 4-5).

A Tabu Search strategy is included in order to enhance feasible solutions. The tabu list $\mathcal{T}$ stores tabu nodes (Line 6), discarding previous solutions. Essentially, the most frequent nodes involved in all solutions after the second phase (VND) are not considered for further solutions during θ iterations, whenever we reach θ^{max} consecutive iterations without improvement. Most frequent nodes are selected if they appear more than ϕ times since the last tabu list refresh. The real numbers ϕ and θ are uniformly chosen at random in the interval $[1, \theta^{max}]$, being θ^{max} a parameter of the algorithm. The specific GRASP phases for the *MCC* and *MEWNC* are described in detail in the following subsections.

3.1. Construction Phase - $Clique$

The construction phase of the proposed algorithm is depicted in Algorithm 2. Let us denote by $\mathcal{C}$ the clique under construction, $\delta_w(U)$ and $\Delta_w(U)$ the highest and lowest weighted-sum in their adjacent

Algorithm 1 HM PSEUDO-CODE

Input: α, θ^{max}, *maxIter*, G
Output: $\mathcal{C}^*$

1: $\mathcal{C}^* \leftarrow \emptyset$
2: $\mathcal{T} \leftarrow \emptyset$
3: **for** *iter = 1* **to** *maxIter* **do**
4: $\quad \mathcal{C} \leftarrow$ CLIQUE*(α, $\mathcal{T}$, G)*
5: $\quad \mathcal{C} \leftarrow$ VND*($\mathcal{C}$, $\mathcal{T}$, G)*
6: $\quad \mathcal{T} \leftarrow$ UPDATE*($\mathcal{T}$, θ^{max}, $\mathcal{C}$)* ▷ Tabu List
7: $\quad$ **if** $|\delta(\mathcal{C})| > |\delta(\mathcal{C}^*)|$ **then**
8: $\quad\quad \mathcal{C}^* \leftarrow \mathcal{C}$
9: **return** $\mathcal{C}^*$

links in the node-set U respectively. The clique $\mathcal{C}$ is initially empty (Line 1), and a multi-start process is considered (Line 2). A Restricted Candidate List (RCL) is defined in Line 3. Observe that the RCL includes nodes with the highest weights in their adjacent links, and α trades greediness for randomization. During the *While* loop of Lines 4-11, a singleton $\{i\}$ is uniformly picked from the RCL (Line 5), and the maximum clique $\mathcal{C}'$ is built using the nodes from the set $\mathcal{C} \cup \{i\}$, specifically, $[\mathcal{C} \cap N(i)] \cup \{i\}$, being $N(i)$ the neighbor-set of node i (see Line 6). The best solution is updated if necessary (Lines 7-8). Observe that the process is finished only if we meet $maxAttempts$ without improvement (Lines 9-11). The reader can appreciate that the output $\mathcal{C}$ is the best feasible clique during the whole process (Line 12).

Algorithm 2 CLIQUE

Input: α, $\mathcal{T}$, $maxAttempts$, G
Output: $\mathcal{C}$

1: $\mathcal{C} \leftarrow \emptyset$
2: $improving = maxAttempts$
3: $RCL \leftarrow \{v \in V - \mathcal{C} : |\delta(v)| \geq \Delta_w(V - \mathcal{C}) - \alpha(\Delta_w(V - \mathcal{C}) - \delta_w(V - \mathcal{C}))\}$
4: **while** $improving > 0$ **do**
5: $\quad i \leftarrow selectRandom(RCL)$
6: $\quad \mathcal{C}' \leftarrow [\mathcal{C} \cap N(i)] \cup \{i\}$
7: $\quad$ **if** $|\delta(\mathcal{C}')| > |\delta(\mathcal{C})|$ **then**
8: $\quad\quad \mathcal{C} \leftarrow \mathcal{C}'$
9: $\quad\quad improving \leftarrow maxAttempts$
10: $\quad$ **else**
11: $\quad\quad improving \leftarrow improving - 1$
12: **return** $\mathcal{C}$

3.2. Local Search Phase - VND

The goal is to combine a rich diversity of neighborhoods in order to obtain an output that is locally optimal solution for every feasible neighborhood. Five neighborhood structures are considered to build a VND Duarte et al. (2016). **Add**, **Swap**, and **Aspiration** are taken from a previous ILS Martins et al. (2015). However, our VND is enriched with 2 additional neighborhood structures, named **Remove** and **Cone**. The following neighborhood take effect whenever the resulting cut-clique is increased:

- **Remove**: a vertex i is removed from a clique $\mathcal{C}$.
- **Add**: if we have a vertex i with $\mathcal{C} \subset N(i)$ we can add i to clique $\mathcal{C}$.
- **Swap**: if we find $j \notin \mathcal{C}$ such that $\mathcal{C} \setminus \{i\} \subseteq N(j)$, we can include j in the clique and delete i.
- **Cone**: generalization of Swap for multiple nodes. The clique $\mathcal{C}$ is replaced by $\mathcal{C} \cup \{i\} \setminus \mathcal{A}$, being i a node that does not belong to $\mathcal{C}$ and $\mathcal{A}$ the nodes from $\mathcal{C}$ that are non-adjacent to i.
- **Aspiration**: this movement offers the opportunity of nodes belonging to the Tabu List to be added.

Observe that the dynamic Tabu list works during the potential additions during **Add**, **Swap** and **Cone**. On the other hand, **Aspiration** provides diversification with an *opportunistic unchoking* process: it picks nodes from the Tabu List instead. For the remaining four local searches, there is an efficient way to determine whether there is an improvement with respect to some neighbor-set. Specifically, the Test Lemmas 4 to 7 are just efficient tests to determine whether there is an improvement under the different local searches **Remove**, **Add**, **Swap** and **Cone** movements, respectively. We call Aspiration Test to Lemma 5 but applied in a different domain (specifically, the candidate nodes must belong to the Tabu List).

The Flow Diagram of our VND is presented in Figure 1. The ordered sequence of local searches are **Remove**, **Add**, **Swap**, **Cone** and **Aspiration** moves. Once an improvement is obtained, the process restarts from the beginning. Observe that, in the output, a locally optimal solution under all neighborhood structures is met.

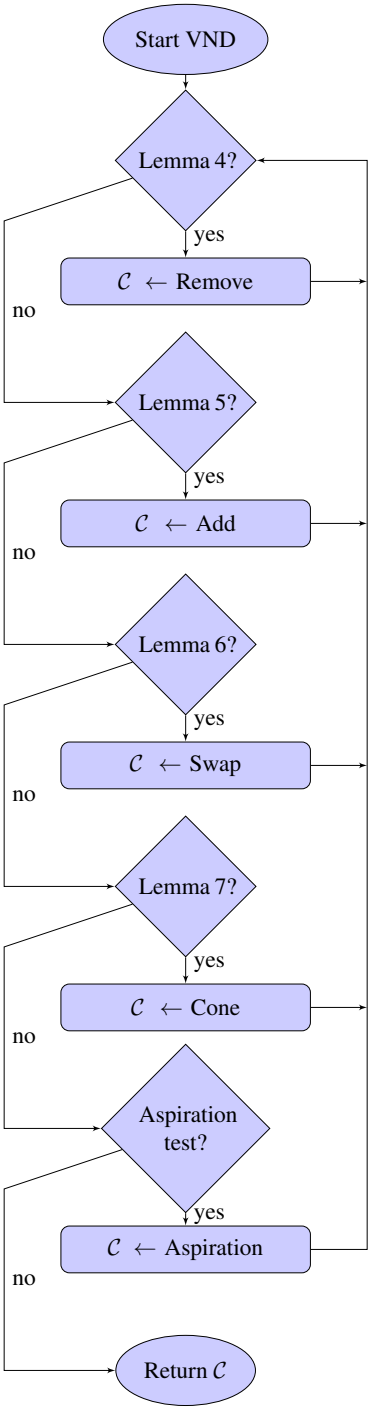


Fig. 1: Flow Diagram for the Local Search Phase - VND.

In the following we re-write the tests for each local search efficiently. We denote $\Omega(v, H)$ to the weighted-sum of all the links between v and the node-set H.

Lemma 4 (Remove). $|\delta(\mathcal{C} \setminus \{i\})| > |\delta(\mathcal{C})|$ iff $\Omega(i, G \setminus \mathcal{C}) < \Omega(i, \mathcal{C})$.

Proof. $|\delta(\mathcal{C} \setminus \{i\})| = |\delta(\mathcal{C})| + \Omega(i, \mathcal{C}) - \Omega(i, G \setminus \mathcal{C}) > |\delta(\mathcal{C})|$; the last inequality holds iff $\Omega(i, G \setminus \mathcal{C}) < \Omega(i, \mathcal{C})$. □

Lemma 5 (Add). $|\delta(\mathcal{C} \cup \{i\})| > |\delta(\mathcal{C})|$ iff $\Omega(i, \mathcal{C}) < \Omega(i, G \setminus \mathcal{C})$.

Proof. $|\delta(\mathcal{C} \cup \{i\})| = |\delta(\mathcal{C})| - \Omega(i, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C}) > |\delta(\mathcal{C})|$; the last inequality holds iff $\Omega(i, \mathcal{C}) < \Omega(i, G \setminus \mathcal{C})$. □

Lemma 6 (Swap). $|\delta(\mathcal{C} \setminus \{j\} \cup \{i\})| > |\delta(\mathcal{C})|$ iff
$\Omega(j, G \setminus \mathcal{C}) + \Omega(i, \mathcal{C}) < \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C})$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} \setminus \{j\} \cup \{i\})| &= |\delta(\mathcal{C})| - \Omega(j, G \setminus \mathcal{C}) + \Omega(j, \mathcal{C}) - \Omega(i, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C}) \\ &= |\delta(\mathcal{C})| - \Omega(j, G \setminus \mathcal{C}) - \Omega(i, \mathcal{C}) + \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C}) \\ &> |\delta(\mathcal{C})|, \end{aligned}$$

where the last inequality holds iff $\Omega(j, G \setminus \mathcal{C}) + \Omega(i, \mathcal{C}) < \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C})$. □

Lemma 7 (Cone). $|\delta(\mathcal{C} \setminus \mathcal{A} \cup \{i\})| > |\delta(\mathcal{C})|$ iff $\sum_{j \in A} \Omega(j, G \setminus \mathcal{C}) + \Omega(i, \mathcal{C}) < \sum_{j \in A} \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C})$.

Proof.

$$\begin{aligned} |\delta(\mathcal{C} \setminus \mathcal{A} \cup \{i\})| &= |\delta(\mathcal{C})| - \sum_{j \in A} \Omega(j, G \setminus \mathcal{C}) + \sum_{j \in A} \Omega(j, \mathcal{C}) - \Omega(i, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C}) \\ &= |\delta(\mathcal{C})| - \sum_{j \in A} \Omega(j, G \setminus \mathcal{C}) - \Omega(i, \mathcal{C}) + \sum_{j \in A} \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C}) \\ &> |\delta(\mathcal{C})| \end{aligned}$$

where the last inequality holds iff $\sum_{j \in A} \Omega(j, G \setminus \mathcal{C}) + \Omega(i, \mathcal{C}) < \sum_{j \in A} \Omega(j, \mathcal{C}) + \Omega(i, G \setminus \mathcal{C})$. □

4. Exact Method for the *MEWNC*

In this section, we present an exact method based on a mathematical formulation. Due to combinatorial nature, we addressed it by integer programming, using the following decision variables:

Given a graph $G = (V, E)$, we denote $\mathcal{C}$ as a set of nodes whose elements define the clique that has the max cut (the subgraph of G induced by $\mathcal{C}$ is the clique that we are looking for).

$$u_i = \begin{cases} 1 & \text{if node } i \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}, \forall i \in V$$

$$w_{(i,j)} = \begin{cases} 1 & \text{if nodes } i, j \in \mathcal{C} \\ 0 & \text{otherwise} \end{cases}, \forall (i,j) \in E$$

An integer programming model is presented below. As in Martins and Gouveia (2015), the constraint that guarantee a clique are from (1) to (5). Constraints in (1) state that both nodes i, j belong to the clique $\mathcal{C}$ if and only if the edge (i, j) belongs to its edges set. Constraint (2) states that if $i, j \in \mathcal{C}$ necessarily (i, j) belongs to its edges set. Constraint (3) states that if an edge does not belong to E, at most one of the nodes belongs to the clique $\mathcal{C}$. Constraints (4) and (5) just state that w_i and $w_{(i,j)}$ are binary variables. Recall that Theorem 6 provides a feasible interval for the size of the clique, c_{min}. Constraints (6) and (7) determine lower and upper bounds LB and UB for the size of the clique, which helps to reduce the space of solutions. The upper bound (Constraint (7)) was proposed by Martins and Gouveia (2015). Our bounding scheme provides a further reduction of the set of candidate solutions.

The goal is to maximize the weighted cut-clique. Let us define the weight of a given node, i, as the weighted-sum of all its adjacent links, $d_i = \sum_{j \in Neighbors(i)} weight(i, j)$. Observe that the weighted cut-clique is the weighted-sum of its nodes, minus twice the weight of its internal links:

$$\begin{array}{lll}
\max & \sum_{i \in V} d_i \times u_i - 2 \times \sum_{(i,j) \in E} w_{(i,j)} \times weight(i,j) & \\
\text{s.a.} & w_{(i,j)} \leq u_i,\ w_{(i,j)} \leq u_j & \forall (i,j) \in E \quad (1) \\
& u_i + u_j \leq w_{(i,j)} + 1 & \forall (i,j) \in E \quad (2) \\
& u_i + u_j \leq 1 & \forall (i,j) \notin E \quad (3) \\
& w_{(i,j)} \in \{0,1\} & \forall (i,j) \in E \quad (4) \\
& u_i \in \{0,1\} & \forall i \in V \quad (5) \\
& \sum_{i \in V} u_i \geq LB & (6) \\
& \sum_{i \in V} u_i \leq UB & (7)
\end{array}$$

5. Computational Results

In order to test the performance of the GRASP/VND algorithm we carried out a fair comparison with respect to the ILP model implemented using IBM CPLEX 12.8. Both are executed on a PC (Intel Core i7, 2.4 GHz, 8GB RAM). The graphs under study were obtained from the SteinLib [1], DIMACS, NIP class with origin in Förster et al. (2003) and the RTN class with origin in Batagelj and Mrvar (2009). The original DIMACS instances do not include weights on its edges. However, we followed the weighting strategy proposed by Pullan (2008), setting $c_{ij} = (i + j)\ mod\ 200 + 1$ in order to study c-fat instances.

Table 1 reports the performance of the GRASP/VND and ILP for each instance considering all links with unit weights (*MCC*). Table 3 reports the same considering instances' original weights. All GRASP/VND algorithm instances were tested using a single run with one-hundred iterations and $\alpha = \frac{1}{2}$, $maxAttempts = \lfloor |V|/10 \rfloor$, $\theta^{max} = 4$. Lower and upper bounds LB and UB were obtained for each topology under study combining Corollary 1 and Theorem 6 for the unweighted and weighted versions of the problem respectively. An exact CPLEX resolution for the ILP was executed with and without Constraint (6), in order to evaluate the performance of our bounding scheme.

The values remarked using bold letters from column $|\delta(\mathcal{C})|$ indicate that the best solution was reached according to the output from the ILP solver with the case of using both bounds (LB and UB).

Following the terminology, $|\delta(\mathcal{C})|$, $|\mathcal{C}|$ and $Time$ represent maximum cut-clique size found, best solution, and the CPU time in seconds for the best solution found. LB, UB columns are reported for the ILP solver which represents lower an upper bounds for the ILP model. Under ILP, $Time$ gives the time to reach the optimum value or the best lower bound to the optimum when the optimum is not attained within the given time limit (10800 seconds).

The reader can appreciate from Tables 1 and 3 that our GRASP/VND algorithm meets the best solution in almost all cases. A global optimum for almost all the instances under study is formally proved using the ILP formulation. Furthermore, our GRASP/VND approach presents consistently smaller CPU times for graphs with large size. The reader can appreciate that Constraint (6) is really worth for the reduction of the computational times.

Table 2 and 4 show the performance of the VND algorithm for *MCC* and *MEWNC* respectively. The activity of every single local search is studied. Swap and Add movements show to be more effective, while Remove and Cone take effect few times. Aspiration has no effect, but it works for dense graphs.

In order to understand the global effectiveness of our VND scheme, a *mid-point test* is performed. The columns Remove, Add, Swap, Cone and Aspiration show the percentage of each kind of movement applied over one-hundred executions of the VND local search phase. The column #moves states the number of moves applied during these iterations. The column entitled mp displays the average gap in percentage between the best solution found in each local search phase with respect to the feasible solution obtained from the construction phase over one-hundred iterations. The reader can appreciate that the VND effect is notorious, since the cut-clique is roughly half an optimum in most cases using only the Construction Phase.

It is worth to remark that we further studied the performance of our GRASP/VND methodology versus a state-of-the-art ILS heuristic for the *MCC*, reported in Martins et al. (2015) and Martins and Gouveia (2015) for the *MEWNC*. We found optimality under all the reported instances which previously

[1]The dataset can be found in the URL http://steinlib.zib.de/steinlib.php

Instances			GRASP/VND			Bounds		ILP with UB			ILP with LB & UB		
name	n	Density	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	LB	UB	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)
i080-001	80	0.039	**13**	2	0.7120	2	8	**13**	2	1.02	**13**	2	0.66
i080-002	80	0.039	**13**	2	1.4779	2	9	**13**	2	0.97	**13**	2	0.62
i080-011	80	0.11	**38**	4	3.0396	3	16	**38**	4	1.38	**38**	4	0.94
i080-044	80	0.2	**80**	5	1.1495	4	26	**80**	5	1.32	**80**	5	0.71
i080-045	80	0.2	**74**	4	0.5748	4	25	**74**	4	1.26	**74**	4	0.90
i080-111	80	0.11	**35**	4	0.4141	3	15	**35**	4	1.33	**35**	4	0.88
i080-112	80	0.11	**39**	3	0.3191	3	19	**39**	3	0.98	**39**	3	0.64
i080-131	80	0.05	**16**	2	1.1908	2	10	**16**	2	1.27	**16**	2	0.73
i080-132	80	0.05	**15**	3	0.5146	2	9	**15**	3	1.19	**15**	3	0.69
i080-142	80	0.2	**74**	4	1.0306	4	25	**74**	4	0.98	**74**	4	0.70
i080-143	80	0.2	**80**	4	0.8546	4	26	**80**	4	1.16	**80**	4	0.85
i160-001	160	0.019	**15**	2	0.6351	2	10	**15**	2	4.02	**15**	2	2.84
i160-002	160	0.019	**14**	2	2.8054	2	9	**14**	2	4.17	**14**	2	2.44
i160-011	160	0.064	**44**	3	4.6177	3	20	**44**	3	37.2	**44**	3	24.53
i160-044	160	0.2	**180**	5	5.7298	5	47	**180**	5	19.8	**180**	5	11.23
i160-045	160	0.2	**173**	5	3.8451	5	42	**173**	5	26.43	**173**	5	14.92
i160-111	160	0.064	**50**	4	5.9956	4	18	**50**	4	8.75	**50**	4	5.59
i160-112	160	0.064	**46**	4	0.47	3	18	**46**	4	7.43	**46**	4	5.71
i160-131	160	0.025	**19**	3	2.5908	3	10	**19**	3	5.01	**19**	3	2.80
i160-132	160	0.025	**22**	3	2.3052	3	12	**22**	3	4.7	**22**	3	2.95
i160-142	160	0.2	**183**	5	3.7192	5	45	**183**	5	23.48	**183**	5	10.63
i160-143	160	0.2	**170**	5	05.097	5	44	**170**	5	19.33	**170**	5	10.57
mc11	400	0.0095	**6**	2	0.7292	2	2	**6**	2	9.88	**6**	2	6.43
c-fat200-1	200	0.077	**81**	9	0.1860	9	17	**81**	9	83.7	**81**	9	29.57
c-fat200-2	200	0.163	**306**	17	0.8388	17	34	**306**	17	201.6	**306**	17	88.79
c-fat200-5	200	0.426	**1892**	43	13.0593	43	86	**1892**	43	4627.9	**1892**	43	1717.39
c-fat500-1	500	0.036	**110**	10	4.77459	10	20	**110**	10	2781.2	**110**	10	1198.31
c-fat500-2	500	0.073	**380**	19	14.1875	19	38	$\geq$380	19	10800	**380**	19	1822.08
c-fat500-5	500	0.186	2304	48	121.32	48	95	$\geq$2304	48	10800	$\geq$2304	48	10800
c-fat500-10	500	0.374	8930	94	33.298	94	188	$\geq$8930	94	10800	$\geq$8930	94	10800

Table 1: GRASP/VND versus ILP for the *MCC*.

achieved optimality. Furthermore, we found the best feasible solutions so far in the remaining cases, with identical results offered by Martins et al. (2015) and Martins and Gouveia (2015) as the reader can see in Section 6.

Instances			GRASP/VND						
name	n	Density	Remove (%)	Add (%)	Swap (%)	Cone (%)	Aspiration (%)	#moves	mp (%)
i080-001	80	0.039	0	38	57	5	0	154	30.949
i080-002	80	0.039	3	32	59	7	0	147	27.571
i080-011	80	0.11	1	48	48	3	0	158	18.647
i080-044	80	0.2	0	54	46	0	0	239	17.390
i080-045	80	0.2	0	56	44	0	0	217	20.585
i080-111	80	0.11	0	59	39	3	0	176	22.572
i080-112	80	0.11	0	48	37	5	0	244	16.261
i080-131	80	0.05	1	30	60	9	0	151	25.983
i080-132	80	0.05	0	32	68	0	0	136	23.402
i080-142	80	0.2	0	50	50	0	0	202	19.762
i080-143	80	0.2	0	52	48	0	0	253	18.467
i160-001	160	0.019	0	22	73	5	0	143	29.452
i160-002	160	0.019	0	20	80	0	0	135	26.427
i160-011	160	0.064	0	57	40	3	0	186	21.803
i160-044	160	0.2	0	60	40	0	0	251	18.079
i160-045	160	0.2	0	53	47	0	0	211	16.358
i160-111	160	0.064	0	61	38	1	0	181	23.816
i160-112	160	0.064	0	55	45	0	0	154	23.940
i160-131	160	0.025	0	39	58	2	0	168	23.710
i160-132	160	0.025	0	42	54	3	0	179	26.588
i160-142	160	0.2	0	57	42	0	0	250	18.146
i160-143	160	0.2	0	59	41	0	0	196	17.664
mc11	400	0.0095	0	100	0	0	0	46	50.000
c-fat200-1	200	0.077	0	100	0	0	0	376	13.159
c-fat200-2	200	0.163	0	100	0	0	0	475	6.338
c-fat200-5	200	0.426	0	99	1	0	0	138	0.505
c-fat500-1	500	0.036	0	100	0	0	0	391	12.965
c-fat500-2	500	0.073	0	100	0	0	0	278	3.736
c-fat500-5	500	0.186	2	98	0	0	0	132	3.252
c-fat500-10	500	0.374	0	100	0	0	0	11	0.132

Table 2: Performance of the Local Search Phase of the GRASP/VND for *MCC*.

Instances			GRASP/VND			Bounds		ILP with UB			ILP with LB & UB		
name	n	Density	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	LB	UB	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)
i080-001	80	0.0380	**1912**	2	0.8522	2	8	**1912**	2	2.06	**1912**	2	0.77
i080-011	80	0.1108	**5468**	3	85.0657	2	16	**5468**	3	2.51	**5468**	3	0.94
i080-044	80	0.2000	**9442**	5	16.7738	3	26	**9442**	5	1.91	**9442**	5	0.97
i080-131	80	0.0506	**2534**	3	5.7053	2	10	**2534**	3	2.12	**2534**	3	0.79
i160-001	160	0.0189	**1691**	2	9.4890	2	10	**1691**	2	7.6	**1691**	2	3.12
i160-011	160	0.0638	**5448**	3	11.1498	2	20	**5448**	3	12.46	**5448**	3	4.01
i160-044.	160	0.2000	**22737**	5	79.3854	3	47	**22737**	5	21.64	**22737**	5	12.37
i160-131	160	0.0252	**2301**	3	19.1936	2	10	**2301**	3	4.21	**2301**	3	2.96
i320-001	320	0.0094	**2274**	2	11.3182	2	10	**2274**	2	15.43	**2274**	2	8.75
i320-011	320	0.0361	**5595**	3	290.534	2	21	**5595**	3	69.89	**5595**	3	26.31
i320-044	320	0.2000	**56366**	5	223.0667	4	90	**56366**	5	1383.2	**56366**	5	441.57
i320-131	320	0.0125	**2396**	2	15.8872	2	10	**2396**	2	9.59	**2396**	2	6.86
i640-001	640	0.0047	**2797**	2	160.0125	2	13	**2797**	2	72.85	**2797**	2	27.43
i640-011	640	0.0202	**6480**	3	50.1892	3	28	**6480**	3	241.11	**6480**	3	67.12
i640-044	640	0.2000	103453	7	4255.319	4	161	$\geq$103453	7	10800	$\geq$103453	7	10800
i640-131	640	0.0063	**2612**	2	27.292	2	12	**2612**	2	101.77	**2612**	2	36.26
mc11	400	0.0095	**1374**	2	25.0436	2	4	**1374**	2	10.8	**1374**	2	6.03
c01	500	0.0050	**92**	2	221.738	2	12	**92**	2	123.41	**92**	2	16.92
c06	500	0.0080	**129**	3	32.8086	2	13	**129**	3	31.42	**129**	3	7.28
c11	500	0.0200	**232**	3	498.365	2	22	**232**	3	154.56	**232**	3	30.35
c16	500	0.1002	**1464**	5	7046.529	4	71	**1464**	5	1341.8	**1464**	5	692.63
d01	1000	0.0025	**102**	3	86.4624	2	13	**102**	3	173.51	**102**	3	69.84
d06	1000	0.0040	**126**	2	245.2698	2	12	**126**	2	230.22	**126**	2	72.56
d11	1000	0.0100	**289**	3	1052.1850	3	23	**289**	3	726.68	**289**	3	77.92
d16	1000	0.0501	**1386**	4	3368.4034	4	73	**1386**	4	6124.6	**1386**	4	2861.09
e01	2500	0.001	**137**	2	3932.207	2	14	**137**	2	947.30	**137**	2	506.30
e06	2500	0.0016	**142**	2	3251.8594	2	16	**142**	2	2312.2	**142**	2	495.11
e11	2500	0.0040	**253**	3	3289.2753	2	24	**253**	3	3855.37	**253**	3	1179.18
e16	2500	0.0200	1252	4	10191.045	3	75	$\geq$1252	4	10800	$\geq$1252	4	10800
c-fat200-1	200	0.077	**9952**	8	77.1055	5	17	**9952**	8	12.52	**9952**	8	11.43
c-fat200-2	200	0.163	**32976**	16	89.67	9	34	**32976**	16	342.37	**32976**	16	43.02
c-fat200-5	200	0.426	204714	40	196.81	24	86	$\geq$202298	38	10800	$\geq$204714	40	10800
c-fat500-1	500	0.036	**14036**	9	350.58	6	20	**14036**	9	1041.2	**14036**	9	630
c-fat500-2	500	0.073	**47798**	17	75484.98	11	38	**47798**	17	8322.7	**47798**	17	4453
c-fat500-5	500	0.186	251324	42	4584.046	25	95	$\geq$241245	45	10800	$\geq$251324	42	10800
c-fat500-10	500	0.37	975450	83	10401.35	48	188	$\geq$788250	126	10800	$\geq$975450	83	10800

Table 3: GRASP/VND versus ILP for the *MEWNC*.

Instances			GRASP/VND						
name	n	Density	Remove (%)	Add (%)	Swap (%)	Cone (%)	Aspiration (%)	#moves	mp (%)
i080-001	80	0.0380	0.03	0.45	0.38	0.14	0.00	29	0.26368
i080-011	80	0.1108	0.00	0.26	0.48	0.26	0.01	524	0.03808
i080-044	80	0.2000	0.00	0.54	0.41	0.06	0.00	235	0.15235
i080-131	80	0.0506	0.06	0.33	0.61	0.00	0.00	18	0.26374
i160-001	160	0.0189	0.00	0.14	0.86	0.00	0.00	14	0.12890
i160-011	160	0.0638	0.00	0.64	0.36	0.00	0.00	11	0.28564
i160-044	160	0.2000	0.00	0.53	0.43	0.04	0.00	246	0.18027
i160-131	160	0.0252	0.00	0.28	0.59	0.12	0.00	32	0.15454
i320-001	320	0.0094	0.00	0.19	0.72	0.08	0.00	36	0.24908
i320-011	320	0.0361	0.00	0.20	0.80	0.00	0.00	35	0.12719
i320-044	320	0.2000	0.00	0.44	0.41	0.15	0.00	34	0.11657
i640-001	640	0.0047	0.00	0.17	0.70	0.12	0.00	40	0.22110
i640-011	640	0.0202	0.00	0.24	0.65	0.12	0.00	17	0.13911
i640-044	640	0.2000	0.00	0.53	0.46	0.01	0.00	256	0.14234
i640-131	640	0.0063	0.00	0.00	0.82	0.18	0.00	11	0.16355
mc11	400	0.0095	0.00	0.33	0.67	0.00	0.00	33	0.14887
c01	500	0.0050	0.00	0.13	0.80	0.07	0.00	15	0.14679
c06	500	0.0080	0.00	0.23	0.77	0.00	0.00	13	0.23910
c11	500	0.0200	0.00	0.27	0.64	0.09	0.00	33	0.17143
c16	500	0.1002	0.00	0.51	0.49	0.00	0.00	774	0.16977
d01	1000	0.0025	0.00	0.31	0.69	0.00	0.00	29	0.20144
d06	1000	0.0040	0.00	0.17	0.83	0.00	0.00	42	0.19271
d11	1000	0.0100	0.00	0.40	0.60	0.00	0.00	20	0.16935
d16	1000	0.0501	0.00	0.43	0.57	0.00	0.00	58	0.18596
e01	2500	0.0010	0.00	0.11	0.83	0.06	0.00	36	0.13332
e06	2500	0.0016	0.00	0.06	0.94	0.00	0.00	51	0.12423
e11	2500	0.0040	0.00	0.21	0.74	0.05	0.00	19	0.10851
e16	2500	0.0200	0.00	0.53	0.47	0.00	0.00	138	0.19983
c-fat200-1	200	0.077	0.40	0.60	0.00	0.00	0.00	10	0.02157
c-fat200-2	200	0.1626	0.33	0.62	0.00	0.00	0.04	135	0.04134
c-fat200-5	200	0.426	0.39	0.58	0.00	0.00	0.03	372	0.00657
c-fat500-1	500	0.0357	0.33	0.63	0.00	0.00	0.03	87	0.03155
c-fat500-2	500	0.0733	0.45	0.51	0.00	0.00	0.04	116	0.01004
c-fat500-5	500	0.186	0.32	0.64	0.00	0.00	0.05	546	0.01795
c-fat500-10	500	0.3738	0.35	0.60	0.00	0.00	0.04	788	0.00272

Table 4: Performance of the GRASP/VND for the Local Search Phase for the *MEWNC*.

Instances			Parameters	Best			Average	
name	n	Density	$maxIter$	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	$\|\delta(\mathcal{C})\|$	Time (s)
c-fat200-1	200	0.077	10E1	**81**	9	0.186043	81	0.372932
c-fat200-2	200	0.163	10E1	**306**	17	0.528618	306	0.806601
c-fat200-5	200	0.426	10E1	**1892**	43	2.9922	1892	4.9419
c-fat500-1	500	0.036	10E1	**110**	10	0.528809	110	2.45625
c-fat500-2	500	0.073	10E1	**380**	19	2.90601	380	5.83031
c-fat500-5	500	0.186	10E1	**2304**	48	9.93067	2304	10.8468
c-fat500-10	500	0.374	10E1	**8930**	94	38.0456	8930	65.7376
p_hat300-1	300	0.244	10E2	**789**	8	129.241	787.9	905.13
p_hat300-2	300	0.489	10E2	**4637**	25	8.2183	4636.2	3659.39
p_hat300-3	300	0.744	10E3	**7740**	36	469.674	7756.8	3992.42
p_hat500-1	500	0.253	10E2	**1621**	9	13.722 7	1621	694.2
p_hat500-2	500	0.505	10E2	**11539**	36	16.196	11401.8	723.93
p_hat500-3	500	0.752	10E3	**18859**	50	679.3	18855.5	723.6155
p_hat700-1	700	0.249	10E2	**2606**	11	305.914	2602.1	439.94
p_hat700-2	700	0.498	10E3	**20425**	44	79.981	20425	839.1
p_hat700-3	700	0.748	10E3	**33480**	62	945.01	33468.1	1807.5
p_hat1000-1	1000	0.245	10E3	**3556**	10	216.49	3556	355.9
p_hat1000-2	1000	0.490	10E4	**31174**	46	2124.2	31174	2538.3
p_hat1000-3	1000	0.744	10E4	**53259**	65	2687.4	53256.1	3584.6
p_hat1500-1	1500	0.253	10E3	**6018**	11	399.92	6018	904.85
p_hat1500-2	1500	0.506	10E4	**67486**	65	2482.08	67486	2942.63
p_hat1500-3	1500	0.754	10E4	**112873**	94	11746.5	112872.1	23162.2
keller4	171	0.649	10E2	**1140**	11	9.18	1140	11.8
keller5	776	0.752	10E4	**15184**	27	1956.2	15183.24	1167.64
keller6	3361	0.818	10E5	**159608**	59	26362.1	158423.2	321731.6
c125_9	125	0.899	10E3	**2766**	34	102.391	2766	253.247
c250_9	250	0.899	10E3	**8123**	44	426.18	8123	831.2
c500_9	500	0.901	10E4	**22691**	57	2354.11	22652.4	4469.86
c1000_9	1000	0.901	10E4	**57149**	68	3924.6	56038.7	4125.66
c2000_5	2000	0.500	10E4	**16106**	16	23472.9	16082.1	23472.9
c2000_9	2000	0.900	5x10E4	**136769**	79	37472.9	135001.2	45472.9
c4000_5	4000	0.500	5x10E4	**36174**	18	31196.2	35891.5	38119.2
MANN_a9	45	0.927	10E3	**412**	16	4.32143	412	145.675
MANN_a27	378	0.990	10E4	**31284**	126	309.746	31244.1	548.54
MANN_a45	1035	0.996	5x10E4	**236406**	344	46881.1	235072.1	52112.36
MANN_a81	3321	0.999	5x10E4	**2436894**	1098	73213.6	2433624.7	96743.12
d1-RTN	2418	0.0032	10E2	**1273**	8	3.834	1273	4.903
d3-RTN	4755	0.0024	10E2	**3526**	12	3.961	3526	5.162
d7-RTN	6511	0.0021	10E2	**5656**	15	4.013	5656	5.348
d15-RTN	7965	0.0020	10E2	**7772**	16	3.967	7772	5.664
d30-RTN	10101	0.0018	10E2	**13099**	21	6.012	13099	6.399
d66-RTN	13308	0.0017	10E2	**22379**	28	6.723	22379	7.403

Table 5: Results of the algorithm for the *MCC* problem

Instances			Parameters	Best			Average	
name	n	Density	$maxIter$	$\|\delta(\mathcal{C})\|$	$\|\mathcal{C}\|$	Time (s)	$\|\delta(\mathcal{C})\|$	Time (s)
d1-RTN	2418	0.0032	10E2	**17151**	8	5.726	17151	6.1
d3-RTN	4755	0.0024	10E2	**28254**	11	5.832	28254	7.025
d7-RTN	6511	0.0021	10E2	**37771**	13	6.675	37771	7.873
SC-NIP-m-t1	991	0.0085	10E2	**2022**	9	4.93	2022	6.286
SC-NIP-r-t1	1394	0.0579	10E3	**20843**	97	12.93	20843	15.88
SC-NIP-r-t2	1394	0.0183	10E3	**8516**	61	9.063	8516	11.21

Table 6: Results of the algorithm for the *MEWNC* problem

6. Comparation with the state-of-the-art

In order to test the performance of the algorithms with a state-of-the-art solution, a fair comparison with respect to an Iterated Local Search offered by Martins et al. (2015) is carried out using DIMACS benchmark and a family of strong sparse graphs known as Reuters Terror News Networks [2]. An analogous comparison with the results obtained by Martins and Gouveia (2015) is carried out, using RTN and SCi-NIP (Saccharomyces cerevisiae metabolic networks (Förster et al., 2003)).

Tables 5 and 6 show the performance of our GRASP/VND algorithm for each instance. All instances were tested using 100 independent runs with $\alpha = \frac{1}{2}$, $maxAttempts = \lfloor |V|/10 \rfloor$ and $\theta^{max} = 10$. The values remarked using bold letters from column $|\delta(\mathcal{C})|$ indicate that the best solution known was previously reached. The parameter $maxIter$ represents the number of iterations considered in the algorithm; $|\delta(\mathcal{C})|$, $|\mathcal{C}|$ and $Time$ represents, respectively, the maximum cut-clique size, best solution and CPU time for the best solution. The same columns are reported for the averaging over 100 independent runs.

The reader can appreciate that our GRASP/VND algorithm meets the best solution known so far in all cases. On one hand GRASP/VND is a more powerful solution than ILS, since the local search from the latter are strictly included in the former. On the other hand, the computational effort is increased in our proposal. Even though a global optimum is not formally proved for some instances, the null gap between ILS and our solution reinforces the evidence of optimality.

The results described in this section reflect that our GRASP/VND methodology is competitive with state-of-the-art solutions for both *MCC* and *MEWNC*. We underscore the simplicity of implementation conducted by simple building blocks (solution construction procedures and local search methods).

[2]The dataset is available in the link: http://vlado.fmf.uni-lj.si/pub/networks/data/CRA/terror.htm

7. Conclusions and Trends for Future Work

Several business models can be represented by Market Basket Analysis (MBA). A relevant marketing approach is to find a subset of items that are strongly correlated with the others. This intuition is formalized by means of a combinatorial optimization problem, called the maximum edge-weight neighborhood clique (*MEWNC*).

In this paper, the $\mathcal{NP}$-Completeness of both the *MCC* and *MEWNC* are established. This fact promotes the development of heuristics and bounds. As a consequence, we offered bounds for both a globally optimal solution and the size of the minimum cardinality clique with maximum cut. Then, a GRASP/VND methodology enriched with Tabu Search is developed to address the *MCC* and *MEWNC*. A fair comparison with an exact ILP formulation confirms the optimality of our approach for hundreds of nodes. Furthermore, the computational effort is reduced for the heuristic under large-sized graphs. The movements Swap and Add have the largest activity for the instances under study. The experiments show that our GRASP/VND heuristic is competitive with state-of-the-art solutions for the *MCC* and *MEWNC*. Further analysis should be done to determine the best order for the VND in terms of computational efficiency.

As future work we would like to implement our solution into a real-life product-placement scenario. In a first stage, we need historical information to determine the links between pairs of items. The physical location of the items must be determined using a complementary geometrical problem with constraints. The solution could consider multi-constrained clustering in order to include categories for the items, or other Machine Learning techniques to determine profiles for the customers, according to the product under study. After the real implementation, the feedback of sales in a period is a valuable metric of success.

Acknowledgment

This work is partially supported by COST Action 15127 *Resilient communication services protecting end-user applications from disaster-based failures*, Math-AMSUD Raredep *Rare events analysis in multi-component systems with dependent components* and STIC-AMSUD ACCON, *Algorithms for the capacity crunch problem in optical networks*.

References

Agrawal, R., Imieliński, T., Swami, A., 1993. Mining association rules between sets of items in large databases. *SIGMOD Rec.* 22, 2, 207–216.

Aguinis, H., Forcum, L.E., Joo, H., 2013. Using market basket analysis in management research. *Journal of Management* 39, 7, 1799–1824.

Amuthan, A., Thilak, K.D., 2016. Survey on tabu search meta-heuristic optimization. In *2016 International Conference on Signal Processing, Communication, Power and Embedded System (SCOPES)*, pp. 1539–1543.

Bader, G.D., Hogue, C.W.V., 2003. An automated method for finding molecular complexes in large protein interaction networks. *BMC Bioinformatics* 4, 2.

Batagelj, V., Mrvar, A., 2009. Pajek datasets (2006).

Brohée, S., van Helden, J., 2006. Evaluation of clustering algorithms for protein-protein interaction networks. *BMC Bioinformatics* 7, 1, 488.

Bruinsma, G., Bernasco, W., 2004. Criminal groups and transnational illegal markets. *Crime, Law and Social Change* 41, 1, 79–94.

Cascio, W.F., Aguinis, H., 2008. Research in industrial and organizational psychology from 1963 to 2007: Changes, choices, and trends. *Journal of Applied Psychology* 93, 5, 1062–1081.

Cook, S.A., 1971. The complexity of theorem-proving procedures. In *Proceedings of the third annual ACM symposium on Theory of computing*, ACM, New York, NY, USA, pp. 151–158.

Duarte, A., Mladenović, N., Sánchez-Oro, J., Todosijević, R., 2016. Variable Neighborhood Descent, Springer International Publishing, Cham.

Förster, J., Famili, I., Fu, P., Palsson, B.Ø., Nielsen, J., 2003. Genome-scale reconstruction of the saccharomyces cerevisiae metabolic network. *Genome research* 13, 2, 244–253.

Garey, M.R., Johnson, D.S., 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman and Company, New York, NY, USA.

Glover, F., Laguna, M., 1997. *Tabu Search*. Kluwer Academic Publishers, Norwell, MA, USA.

Henzinger, M., Lawrence, S., 2004. Extracting knowledge from the world wide web. *Proceedings of the National Academy of Sciences* 101, suppl 1, 5186–5191. http://www.pnas.org/content/101/suppl_1/5186.full.pdf.

Hüffner, F., Komusiewicz, C., Moser, H., Niedermeier, R., 2008. Enumerating isolated cliques in synthetic and financial networks. In Yang, B., Du, D.Z. and Wang, C.A. (eds), *Combinatorial Optimization and Applications*, Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 405–416.

Karp, R.M., 1972. Reducibility among combinatorial problems. In Miller, R.E. and Thatcher, J.W. (eds), *Complexity of Computer Computations*. Plenum Press, pp. 85–103.

Martins, P., 2012. Cliques with maximum/minimum edge neighborhood and neighborhood density. *Computers & Operations Research* 39, 3, 594–608.

Martins, P., Gouveia, L., 2015. Solving the maximum edge-weight clique problem in sparse graphs with compact formulations. *EURO Journal on Computational Optimization* 3, 1–30.

Martins, P., Ladrón, A., Ramalhinho, H., 2015. Maximum cut-clique problem: ILS heuristics and a data analysis application. *International Transactions in Operational Research* 22, 5, 775–809.

Pullan, W., 2008. Approximating the maximum vertex/edge weighted clique using local search. *Journal of Heuristics* 14, 2, 117–134.

Resende, M.G., Ribeiro, C.C., 2016. *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*. Computational Science and Engineering. Springer-Verlag New York.

Salhi, S., 2014. Handbook of metaheuristics (2nd edition). *Journal of the Operational Research Society* 65, 2, 320–320.

Tan, P.N., Steinbach, M., Kumar, V., 2005. *Introduction to Data Mining*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Part II

Uniformly Most-Reliable Graphs

Chapter 5

A Hybrid GRASP/VND Heuristic for the Design of Highly Reliable Networks

In this chapter we develop an ideal VND metaheuristic that returns a uniformly most-reliable graph. Since it has exponential time, we must trade accuracy for computational effort. As a consequence, a full GRASP/VND heuristic is introduced and novel networks that show high reliability are found.

A Hybrid GRASP/VND Heuristic for the Design of Highly Reliable Networks

Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero, Luis Stábile

Facultad de Ingeniería, Universidad de la República. Montevideo, Uruguay.

(mbourel,canale,frobledo,promero,lstabile)@fing.edu.uy

Abstract. There is a strong interplay between network reliability and connectivity theory. In fact, previous studies show that the graphs with maximum reliability, called uniformly most-reliable graphs, must have the highest connectivity. In this paper, we revisit the underlying theory in order to build uniformly most-reliable cubic graphs. The computational complexity of the problem promotes the development of heuristics. The contributions of this paper are three-fold. In a first stage, we propose an ideal Variable Neighborhood Descent (VND) which returns the graph with maximum reliability. This VND works in exponential time. In a second stage, we propose a hybrid GRASP/VND approach that trades quality for computational effort. A construction phase enriched with a Restricted Candidate List (RCL) offers diversification. Our local search phase includes a factor-2 algorithm for an Integer Linear Programming (ILP) model. As a product of our research, we recovered previous optimal graphs from the related literature in the field. Additionally, we offer new candidates of uniformly most-reliable graphs with maximum connectivity and maximum number of spanning trees.

Keywords: Network Optimization, Maximum Reliability, Heuristics, GRASP, VND, ILP.

1 Motivation

In network reliability analysis, the goal is to find the probability of correct operation of a system [6, 2]. The context of the original problem determines our notion of correct operation. For instance, delay sensitive applications such as videoconference require a hop-constrained network, where the terminals should be connected by short paths [5]. Wireless systems deal with a hostile environment with mobility (fading, handover and coverage, among other challenges). The goal is to achieve a Grade of Service (GoS) during the busy hour, and node-reliability analysis is more suitable for this context [12]. The interaction between peers in a cooperative environment suggests potential links, and a link-reliability analysis is adequate for this context. Peer-to-peer systems suffer from starvation when the missing-piece syndrome affect all the system [9]. Clearly, the swarm (or population) should be connected, and the

all-terminal reliability model is a suitable tool in order to understand this phenomena.

Several researchers from different fields of knowledge (mathematics, computer science, engineering), shaped the body of network reliability analysis, given the application and importance of the underlying models. A fundamental problem is to find the connectedness probability of a random graph, subject to link failures, called the *all-terminal reliability*. The scientific literature around this problem is vast; however, this problem is not fully understood yet. The corresponding practical problem is to connect p sites using q links in the *best* way, this is, to find the graph whose all-terminal reliability is maximum among all (p, q)-graphs. Such graphs are called *uniformly most-reliable graphs*.

The main contributions of this paper are the following:

1. An exact VND that returns uniformly most-reliable graphs is presented.
2. A hybrid GRASP/VND heuristic is introduced in order to find graphs with high reliability. It trades quality for computational feasibility.
3. An Integer Lineal Programming (ILP) formulation called Regularity Problem is proposed. The goal is to find a regular graph starting from a non-regular one moving as minimum number of links as possible.
4. A factor 2 for the Regularity Problem is introduced.
5. Novel networks that show high reliability and connectivity are found, as a result of our hybrid heuristic.

The document is organized in the following manner. Section 2 formally states the problem and breakthroughs in the field of uniformly most-reliable graphs. Section 3 presents an exact VND that runs in exponential time, and a hybrid GRASP/VND heuristic that trades quality for computational feasibility. As a product, we offer novel cubic networks with high reliability in Section 4. Concluding remarks and open problems are discussed in Section 5.

2 Uniformly Most-Reliable Graphs

2.1 Definition

In the following, we work with undirected graphs without loops, and a graph with p nodes and q links is a (p, q)-graph.

Definition 1. *Consider a graph G with perfect nodes but independent link failures with identical probability $\rho \in (0, 1)$. The* all-terminal reliability*, $R_G(\rho)$, is the probability that the resulting subgraph remains connected.*

The *unreliability* $U_G(\rho) = 1 - R_G(\rho)$ can be expressed using sum-rule:

$$U_G(\rho) = \sum_{k=0}^{q} m_k(G)\rho^k(1-\rho)^{q-k}, \tag{1}$$

being $m_k(G)$ the number of spanning disconnected subgraphs of G with exactly $q-k$ links. Therefore, $R_G(\rho)$ is a polynomial in $\rho \in (0, 1)$, and its determination is reduced to counting the numbers $\{m_k\}_{k=0,\ldots,q}$.

Definition 2. *A (p,q)-graph H is* uniformly most-reliable *if $R_H(\rho) \geq R_G(\rho)$ for all (p,q)-graph G and all $\rho \in (0,1)$.*

Alternatively, H is uniformly most-reliable if its unreliability $U_H(\rho)$ is dominated (i.e., upper-bounded) by all functions $U_G(\rho)$ for all (p,q)-graph G.

2.2 Breakthroughs

In this section we present fundamental results that are the cornerstone in the theory of uniformly most-reliable graphs. The following section briefly describes the main findings that complement the fundamental results.

In 1977, Arnie Rosenthal formally proved that the K-terminal reliability evaluation belongs to the class of $\mathcal{NP}$-Hard computational problems [19]. The key concept of the proof is the reducibility introduced in 1972 by Richard Karp, which represents a foundational work in computational complexity [13]. As corollary, finding uniformly most-reliable graphs is a hard problem as well.

Observe that if $m_k(H) \leq m_k(G)$ for all $k \in \{0,\ldots,q\}$ and (p,q)-graph G, then H is uniformly most-reliable. This is a simple but elegant interplay between network reliability analysis and connectivity theory. Curiously enough, the converse is still an open problem:

Conjecture 1 (Boesch et. al.). If G is uniformly most-reliable (p,q)-graph, then $m_k(G) \leq m_k(H)$ for all (p,q)-graph H.

If $\lambda(H)$ denotes the connectivity of H and $\tau(H)$ its number of spanning trees, the following necessary criterion holds [1]:

Corollary 1. *A uniformly most-reliable graph H must have the maximum tree-number $\tau(H)$, maximum connectivity $\lambda(H)$, and the minimum number $m_\lambda(H)$.*

Corollary 1 wakes up interest in two special sub-problems: the maximum connectivity and maximum tree-number of a graph. In the second book ever written in graph theory, Claude Berge challenges the readers to find the graph with maximum connectivity among all graphs with a fixed number of nodes and links. Frank Harary provided not only a full answer, but also found connected graphs with minimum and maximum diameter [10]. The idea behind his construction is simple: by handshaking, the average degree of a (p,q)-graph is $\frac{2q}{p}$. Therefore, $\lambda \leq \lfloor \frac{2q}{p} \rfloor$. Harary graphs achieve this upper-bound, which represents the maximum connectivity of a graph.

Gustav Kirchhoff solved linear time-invariant resistive circuits, and as corollary he introduced the Matrix-Tree theorem, where he counts the number of spanning trees of a connected graph (i.e., the tree-number) using the determinant of a matrix [15, 3]. This breakthrough in electrical systems launched the theory of trees, which provides the building blocks in communication design. However, the corresponding extremal problem is not well understood: find the graph with a fixed number of nodes and links that maximizes the tree-number.

For convenience we say that a (p, q)-graph, H, is t-optimal if $\tau(H) \geq \tau(G)$ for every (p, q) graph G. Briefly, Corollary 1 claims that uniformly most-reliable graphs must be t-optimal and max-λ min-m_λ, where λ denotes the edge connectivity.

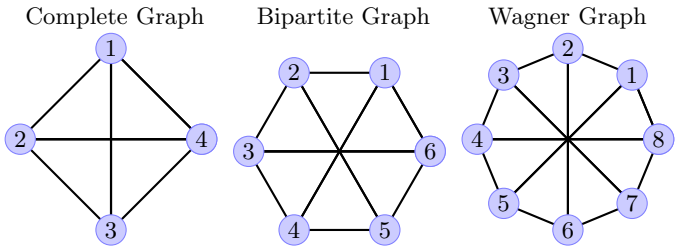
Another breakthrough from the related literature is a reliability improving graph transformation called *swing surgery*, independently discovered by Kelmans [14] and Satyanarayana et. al. [20]. Specifically, if we are given a (p, q)-graph $G = (V, E)$, two nodes $x, y \in V$ with respective neighboring nodes S_x and S_y with $S_x \setminus \{y\} \subset S_y$, $B \subseteq S_y - \{x\}$ such that $B \cap S_x = \emptyset$, and G' the graph obtained by G by removing the links $\{(y, z), z \in B\}$ and adding the links $\{(x, z), z \in B\}$, then $R_{G'}(\rho) \geq R_G(\rho)$ for all $\rho \in (0, 1)$.

2.3 Findings

Redundancy is of paramount importance in communication networks. For that reason, in the following we are specifically focused on uniformly most-reliable cubic graphs (i.e., 3-regular connected graphs). By handshaking, this is the case of $(2r, 3r)$-graphs for some $r \geq 2$. Recall that Möbius graph M_n is precisely the elementary cycle C_{2n} together with all the diameters (opposite nodes are also linked). So far, the findings of uniformly most-reliable cubic graphs can be summarized in the following list:

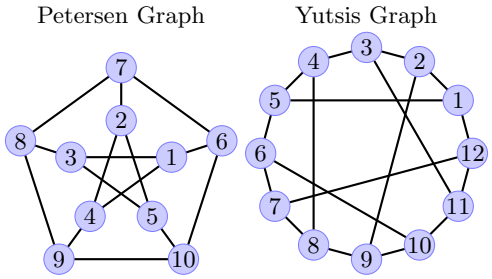
- $K_4 = M_2$; complete graph with 4 nodes; case $r = 2$; see [4].
- $K_{(3,3)} = M_3$; complete bipartite graph; case $r = 3$; see [23].
- $W_4 = M_4$; Wagner graph ($r = 4$); see [18].
- P_5; Petersen graph ($r = 5$); see [16].
- Y_6; Yutsis graph ($r = 6$); see [22].

The reader is invited to consult the corresponding references for a mathematical proof that these graphs are uniformly most-reliable. They are sketched in Figures 1-2. By computational limits, currently it is not possible to find uniformly most-reliable cubic graphs for $r \geq 7$. The goal of this paper is to build highly reliable cubic graphs for $r \in \{7, \ldots, 10\}$, finding a trade-off between quality and computational effort.



(a) $K_4 = M_2$ (b) $K_{(3,3)} = M_3$ (c) M_4

Fig. 1. Möbius graphs M_2, M_3 and M_4



(d) P_5 (e) Y_6

Fig. 2. Petersen P_5 and Yutsis graph Y_6.

2.4 Equivalent Combinatorial Problem

It is worth to remark that the problem of finding uniformly most-reliable graphs is a simultaneous minimization of an uncountable family of numbers $\{U_G(\rho)\}_{\rho\in(0,1)}$. However, if Boesch Conjecture holds, we observe that the problem can be translated to a (single-objective) combinatorial optimization problem. Specifically, let us denote $m(G)$ to the number of disconnected spanning subgraphs for G. By the definition of link disconnecting sets, we get that:

$$m(G) = \sum_{k=0}^{q} m_k(G).$$

Proposition 1. *Consider natural numbers p and q such that there exists a unique (p,q)-graph. If Boesch conjecture holds, then G is uniformly most-reliable (p,q) graph if and only if $m(G)$ is minimum.*

Proof. Assume that G is uniformly most-reliable. By Boesch conjecture, every disconnecting set $m_k(G)$ is minimum among all the other (p,q)-graphs. Therefore, the number $m(G) = \sum_{k=0}^{q} m_k(G)$ is also minimum in this set.
For the converse, consider a (p,q)-graph G such that $m(G)$ is minimum. By hypothesis, there exists some (p,q)-graph, denoted by H. By Boesch conjecture, $m_k(H) \leq m_k(G)$. Since $m(G) = \sum_{k=0}^{q} m_k(G)$ is minimum, the only possibility is that $m_k(G) = m_k(H)$ for all k. Therefore, G and H share the same unreliability polynomial. By uniqueness, we must have $G = H$, and the statement is proved.

In short, Proposition 1 tells us that if Boesch conjecture holds, then finding uniformly most-reliable graphs is equivalent to the minimization of disconnected spanning subgraphs $m(G)$. This result reinforces the evidence that the optimum graphs under connectivity (i.e., purely deterministic) and reliability optics (probabilistic) share common properties.

In this work we are focused on the minimization of disconnecting spanning subgraphs $m(G)$. In this paper, we offer highly reliable cubic graphs, which share strong connectivity properties as well, supported by Proposition 1.

It is well-known that finding the coefficients $\{m_k(G)\}_{k=0,\ldots,q}$ belongs to the hierarchy of $\#\mathcal{P}$-Complete counting problems [21]. Furthermore, the number $m(G) = \sum_{k=0}^{q} m_k(G) = T(1,2)$, is precisely Tutte polynomial evaluated at the point $(1,2)$, which is a $\#\mathcal{P}$-Hard counting problem, even for bipartite planar graphs [11]. Here, we propose a pointwise statistical estimation of this number. Monte Carlo is a noteworthy computational tool for simulation. From a macroscopic point of view, the idea is to faithfully simulate a complex system (or a part of it), and consider N independent experiments of that simulation, in order to determine the performance of the system (or subsystem) and assist decisions on it [8].

We will use Crude Monte Carlo (CMC) in order to provide an unbiased statistical estimation for $m(G)$. First of all, observe that $m(G)$ is strictly related with the unreliability evaluation at $\rho = \frac{1}{2}$:

$$U_G(\frac{1}{2}) = \sum_{k=0}^{q} m_k(G)(\frac{1}{2})^k(\frac{1}{2})^{q-k} = \frac{m(G)}{2^q}. \tag{2}$$

Equation 2 shows that, alternatively, we must minimize $U_G(\frac{1}{2})$, or the probability that the resulting subgraph is disconnected under identical independent link failures with probability $\rho = \frac{1}{2}$. For any given graph G, let us consider a sample of random graphs $G_1, \ldots, G_N$ picked independently with link failures $\rho = \frac{1}{2}$, and independent Bernoulli variables $X_1, \ldots, X_N$ such that $X_i = 1$ if and only if G_i is disconnected. By strong law of large numbers, the mean sample $\overline{X_N}$ converges almost surely to $u = U_G(\frac{1}{2})$. Therefore, in order to decide whether $m(G_1) < m(G_2)$ or not, we use the criterion $\overline{X_N}^1 < \overline{X_N}^2$ for N large enough, being $\overline{X_N}^i$ the mean sample for the graph G_i. This criterion avoids the full determination of the coefficients $m_k(G)$, and it will be useful for the design of a GRASP/VND heuristic to build highly reliable graphs.

3 Metaheuristics

In this section we develop an ideal VND metaheuristic that returns a uniformly most-reliable graph. Since it has exponential time, we must trade accuracy for computational effort. As a consequence, a full GRASP/VND heuristic is introduced.

3.1 VND

Variable Neighborhood Descent (VND) explores several neighborhood structures in a deterministic order. Its success is based on the simple fact that different neighborhood structures do not usually have the same local minima.

Thus, the local optima trap problem is addressed by a deterministic change of neighborhoods [7].

Recall that a simultaneous minimization of the coefficients $\{m_k(G)\}_{k=0,\ldots,q}$ is a sufficient condition for G to be uniformly most-reliable. Therefore, if there is one local search dedicated to each coefficient, the output must be uniformly most-reliable. Trivial neighborhood structures where all (p,q)-graphs are neighbors of some fixed graph work. However, the cardinality of the search-space of (p,q)-graphs is $\binom{p(p-1)/2}{q}$. Therefore, an exhaustive search among the trivial neighborhood structures of all (p,q)-graphs is computationally prohibitive.

3.2 GRASP/VND Heuristic

GRASP is an iterative multi-start process which operates in two phases [17]. In the Construction Phase a feasible solution is built whose neighborhood is then explored in the Local Search Phase [17]. The second phase is usually enriched by means of different variable neighborhood structures, for instance, VND.

We adapt the previous ideal VND in order to obtain a feasible computational solution in a multi-start fashion with diversification in a previous construction phase. Algorithm *HighlyReliable* receives a maximum number of iterations *iter*, a natural number $r \geq 2$, and returns a highly reliable cubic $(2r, 3r)$-graph.

Algorithm 1 $G = HighlyReliable(r, iter)$

```
1: G ← M_r
2: for i = 1 to iter do
3:    G_input ← GreedyRandomized(r, α)
4:    G(i) ← VND(G_input)
5:    if m(G(i)) ≤ m(G) for all k then
6:       G ← G(i)
7:    end if
8: end for
9: return G
```

In Line 1, the graph is initialized in Möbius graph M_r, which is known to be optimal for the cases where $r \in \{2, 3, 4\}$. In a **for**-loop with *iter* iterations (Lines 2-8), we iteratively call in sequence the Construction Phase (Lines 3) and VND (Line 4). If the number of disconnecting spanning subgraphs $m(G(i))$ is dominated by $m(G)$, the current graph G is replaced by $G(i)$ (Lines 5-6). It is worth to remark that the test $m(G(i)) \leq m(G)$ considers the criterion detailed in Subsection 2.4. The best graph among all the iterations is returned as the output (Line 9).

In the following, we provide details of the Construction Phase ($GreedyRandomized(r)$ from Line 3) and Local Search Phase (VND function,

from Line 4). The result is not necessarily a uniformly most-reliable network, but a highly reliable cubic network, which is useful for practical purposes.

Algorithm 2 $G = Construct(r)$

```
1: U ← RandomNumbers(r(r − 1)/2)
2: G ← RandomTree(U)
3: δ ← min_{v∈G}{deg(v)}
4: Δ ← max_{v∈G}{deg(v)}
5: RCL ← {(v_i, v_j) : deg(v_i)deg(v_j) ≤ δ^2 + α(Δ^2 − δ^2)}
6: for i = 1 to r + 1 do
7:     e_i ← Random(RCL)
8:     G ← G ∪ {e_i}
9:     RCL ← Update(RCL, e_i)
10: end for
11: return G
```

Construction Phase The main idea is to start with a random tree with $2r - 1$ links and insert adequately $r + 1$ links meeting a final size of $3r$ links. In Line 1, for every pair of potential links we pick independent numbers in $(0, 1)$ uniformly chosen at random. In this way, we get random costs c_{ij} for every pair of nodes v_i and v_j. A random tree is found in Line 2. Specifically, function $RandomTree$ applies Kruskal algorithm with the costs c_{ij}. The minimum and maximum degree of the resulting graph are found in Lines 3 and 4 respectively. The addition of the remaining $r + 1$ links takes place in the block of Lines 5-10. A Restricted Candidate List, RCL, selects a percentage of α links $e_{(i,j)}$ with the lowest product degrees $deg(v_i) \times deg(v_j)$; see Line 5. In the **for**-loop of Lines 6-10, links are iteratively picked from the RCL (Line 7) and added to the graph G (Line 8). Observe that the RCL should be updated, since the degrees are modified in each iteration. This operation takes place in function $Update(RCL, e_i)$ (Line 9). Clearly, G is not necessarily regular, but the effect of the RCL provides diversity in the solutions. Naturally, it trades greediness for randomization with the parameter α, and tends to return almost-regular graphs.

Local Search Phase In the Local Search Phase, a VND is considered with the following movements:

1. *Surgery*: applies the graph transformation called Swing Surgery (see Subsection 2.2).
2. *Regular*: returns a regular graph after adequate link addition/deletions.
3. *Crossing*: tests whether the tree-number is increased after all feasible graph-crossings.

A graph G is *healthy* if there is no feasible surgery that improves the reliability uniformly in (0,1). In other words, it is locally optimum with respect to local

movements under Swing Surgery. Analogously, we say that G is *strong* if G has the largest tree-number with respect to all feasible crossings (i.e., it is a locally optimum solution with respect to local movements of $Crossing$). Figure 3 presents the full VND. The reader can observe that the output G is regular, healthy and strong.

In the following, we explain the three movements. Let us start with $Surgery$ and $Crossing$ (as we will see, *Regular* movement includes an ILP formulation, a new result on approximation algorithms and an exact polynomial time algorithm to solve it). $Surgery$ just applies a reliability-improving graph transformation called Swing Surgery from Subsection 2.2 whenever possible. Finally, $Crossing$ tries to find an edge-crossing with largest tree-number. Specifically, if the links $e_1 = (x, y)$ and $e_2 = (z, t)$ belong to G, but $(x, z), (y, t)$ do not belong to G, $Crossing$ counts the tree-number of the new graph $G' = (G - \{(x, y), (z, t)\} \cup \{(x, z), (y, t)\}$. The tree-number is efficiently found by Kirchhoff theorem, as any cofactor of the Laplacian matrix [3].

The main idea of *Regular* is to return a regular graph, starting from a non-regular one. *Regular* movement is a solution to an Integer Linear Programming formulation. Consider the input graph $G = (V, E)$, the resulting regular graph $G' = (V, E')$ and the following binary variables:

- $e_{(i,j)} = 1$ iff $(i, j) \in E$ (adjacency matrix for G);
- $a_{(i,j)} = 1$ iff $(i, j) \in E' - E$ (links added to G');
- $r_{(i,j)} = 1$ iff $(i, j) \in E - E'$ (links removed from G).

Our goal is to minimize the number of addition/deletions in order to return a 3-regular graph G'. The Regularity Problem can be formalized by the following ILP:

$$\min \sum_{i<j} a_{(i,j)} \tag{3}$$

$$s.t. \tag{4}$$

$$\sum_{i<j} a_{(i,j)} = \sum_{i<j} r_{(i,j)} \tag{5}$$

$$\sum_{i<j} a_{(i,j)} + e_{(i,j)} - r_{i,j} = 3 \, \forall i \in \{1, \ldots, 2r\} \tag{6}$$

$$a_{(i,j)}; r_{(i,j)} \in \{0, 1\} \, \forall i < j. \tag{7}$$

Constraint 5 states that the number of added/removed links must be identical, so, $|E| = |E'|$. Constraints 6 state that the resulting graph G' must be 3-regular. Finally, Constraints 7 determine the binary domain for the decision variables $a_{(i,j)}$ and $r_{(i,j)}$.
In the following, *Regular* movement is specified.

Algorithm 3 $G = Regular(G)$

1: $\Delta \leftarrow \max_{v \in V(G_{in})}\{deg(v)\}$
2: $\delta \leftarrow \min_{v \in V(G_{in})}\{deg(v)\}$
3: **while** $\delta(G) < \Delta(G)$ **do**
4: $x \leftarrow \arg\max_{u \in V}\{deg(u)\}$
5: $y \leftarrow \arg\min_{u \in V}\{deg(v')\}$
6: $z \leftarrow Random(N(x) - N(y))$
7: $G \leftarrow (G - (x, z)) \cup (y, z)$
8: **end while**
9: **return** G

Regular function receives a (p, q)-graph G and returns a regular graph. The key is to move links from nodes with the highest degree to nodes with the lowest degree. In Lines 1-2, the maximum and minimum degrees for the input graph are found. A **while**-loop (Lines 3-8) takes effect whenever $\delta < \Delta$. Since the degree of some low-degree (high-degree) node is increased (resp. decreased) by a unit in all the iterations, the number of iterations is finite, and the algorithm returns a 3-regular graph. In Lines 4 and 5, we pick some node x (y) with the highest (resp. lowest) degree. Since $deg(x) > deg(y)$, there exists some node, z, such that z is adjacent to x but non-adjacent to y (Line 6). In the iteration, the link (x, z) is deleted, while (y, z) is added (Line 7). Observe that, in the resulting graph, the degree of x (y) is decreased (resp. increased), but the degree of the pivotal node z is identical. The output of **while**-loop must be a regular graph, which is returned in Line 9.

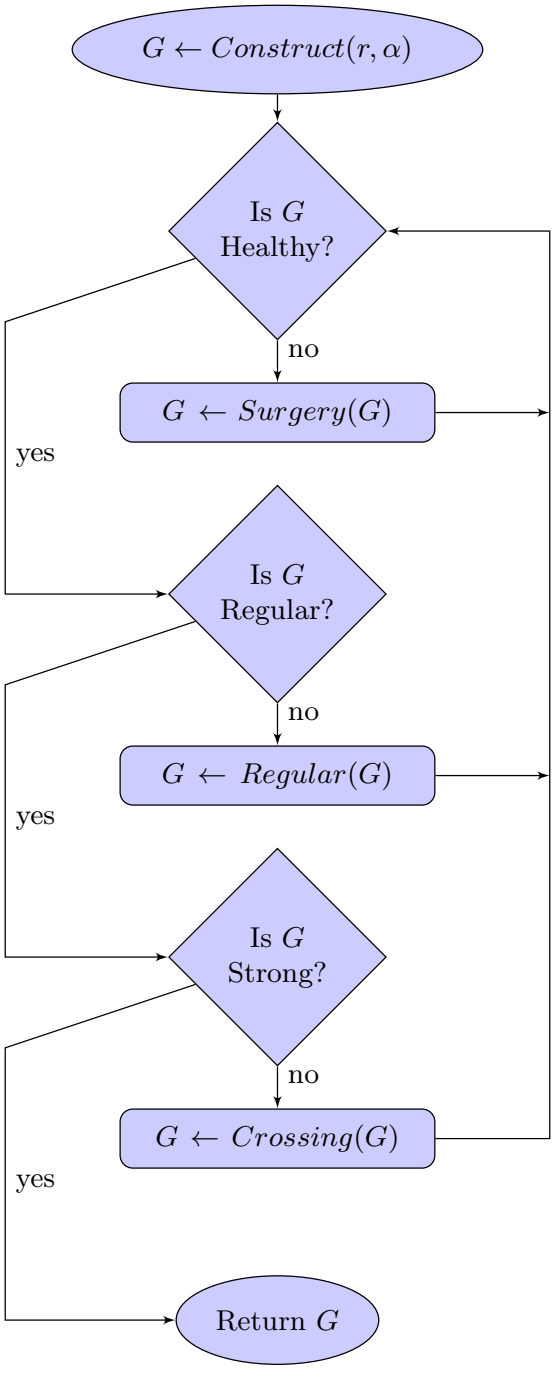


Fig. 3. Flow Diagram for the Local Search Phase - VND.

Theorem 1. *Regular is an approximation algorithm with factor 2 for the Regularity Problem.*

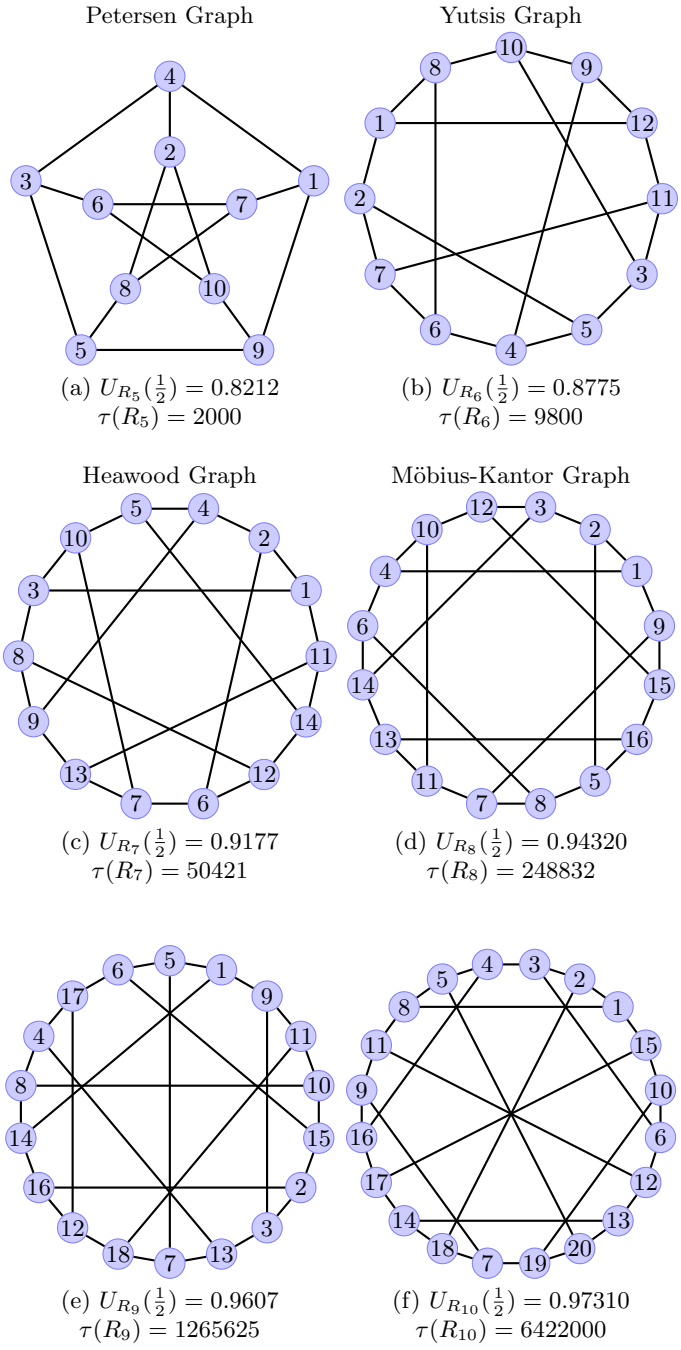
Proof. In an arbitrary link addition/deletion, we can reduce at most 2 degrees of high-degree nodes and increase 2 degrees in the set of low-degree nodes. *Regular* performs methodically an addition/reduction of a single-degree in each movement. Therefore, the number of movements during the execution of *Regular* is, at most, twice the optimal solution. □

4 Results

By construction, it is clear that our GRASP/VND heuristic returns M_r for the known cases $r \in \{2, 3, 4\}$ (see Line 1 of *Construction*). In order to test the effectiveness of our GRASP/VND heuristic, we look for $r \in \{5, \ldots, 10\}$. We know beforehand that the uniformly most reliable graph for $r = 5$ is Petersen graph [16]; when $r = 6$ Yutsis graph Y_6.

Figure 4 sketches the six resulting graphs for $r \in \{5, 6, 7, 8, 9, 10\}$ using $iter = 10^5$ iterations, $\alpha = 0.5$ and sample size $N = 10^4$ for Crude Monte Carlo for the pointwise reliability estimation of $U_G(\frac{1}{2})$ as detailed in Subsection 2.4. Figures 4 (a) and (b) depict Petersen and Yutsis graphs respectively. These base-steps confirm that our hybrid GRASP/VND heuristic is able to discover uniformly most-reliable graphs, as the recent literature in the field confirms with mathematical proofs [16].

Curiously enough, the following cases for $r \geq 7$ are not covered in the related literature. The graphs produced for $r = 7$ and $r = 8$ are strongly symmetric, and they are respectively Heawood and Möbius-Kantor graphs (Figures 4 (c) and (d) depict both graphs). Finally, we could not identify the graphs from Figures 4 (e) and (f) with a previous known name. These results are encouraging, since some uniformly most-reliable graphs were identified. Furthermore, an exhaustive computational test with cubic graphs with girth greater than 3 confirms that the resulting graphs have the maximum tree-number (therefore, they are the only candidates of uniformly most-reliable graphs).



(a) $U_{R_5}(\frac{1}{2}) = 0.8212$
$\tau(R_5) = 2000$

(b) $U_{R_6}(\frac{1}{2}) = 0.8775$
$\tau(R_6) = 9800$

(c) $U_{R_7}(\frac{1}{2}) = 0.9177$
$\tau(R_7) = 50421$

(d) $U_{R_8}(\frac{1}{2}) = 0.94320$
$\tau(R_8) = 248832$

(e) $U_{R_9}(\frac{1}{2}) = 0.9607$
$\tau(R_9) = 1265625$

(f) $U_{R_{10}}(\frac{1}{2}) = 0.97310$
$\tau(R_{10}) = 6422000$

Fig. 4. Results for $r \in \{5, 6, 7, 8, 9, 10\}$

5 Conclusions and Trends for Future Work

Reliability maximization is a relevant problem from network design. Potential applications include virtual and wireless systems, and cooperative environments in a hostile system, where the links may fail.

In the theory of reliability maximization, the goal is to find uniformly most-reliable graphs. The breakthroughs and main result of this theory are here outlined. It has half a century of development; however, there are key questions without concluding answers. Boesch conjecture and Wagner extension are just examples.

Supported by the computational complexity of the problem, we first present a VND metaheuristic that returns all uniformly most-reliable graphs. The main drawback of this proposal is the computational effort. As a consequence, we then develop a hybrid GRASP/VND heuristic that keeps the most meaningful elements of the first VND implementation, enriched with a construction phase in order to gain diversification.

The first results are encouraging. Our hybrid GRASP/VND was able to detect a couple of previous uniformly most-reliable graphs from the related literature. Furthermore, it returns new candidates of such optimal graphs from a reliability viewpoint.

There are several trends for future work. A powerful methodology to find uniformly most-reliable graphs is not known. The power of different graph transformations such as iterative augmentation and swing surgery is not explored yet.

6 Acknowledgements

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos*.

References

1. D. Bauer, F. Boesch, C. Suffel, and R. Van Slyke. On the validity of a reduction of reliable network design to a graph extremal problem. *IEEE Transactions on Circuits and Systems*, 34(12):1579–1581, 1987.
2. L.W. Beineke, R.J. Wilson, and O.R. Oellermann. *Topics in Structural Graph Theory*. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2012.
3. N. Biggs. *Algebraic Graph Theory*. Cambridge Mathematical Library. Cambridge University Press, 1993.
4. F. T. Boesch, X. Li, and C. Suffel. On the existence of uniformly optimally reliable networks. *Networks*, 21(2):181–194, 1991.

5. Eduardo Canale, Hctor Cancela, Franco Robledo, Pablo Romero, and Pablo Sartor. Full complexity analysis of the diameter-constrained reliability. *International Transactions in Operational Research*, 22(5):811–821, 2015.
6. Charles J. Colbourn. Reliability issues in telecommunications network planning. In *Telecommunications network planning, chapter 9*, pages 135–146. Kluwer Academic Publishers, 1999.
7. Abraham Duarte, Nenad Mladenović, Jesús Sánchez-Oro, and Raca Todosijević. *Variable Neighborhood Descent*, pages 1–27. Springer International Publishing, Cham, 2016.
8. George S. Fishman. *Monte Carlo: Concepts, Algorithms and Applications*. Springer Verlag, New York, NY, USA, 1996.
9. B. Hajek and J. Zhu. The missing piece syndrome in peer-to-peer communication. In *2010 IEEE International Symposium on Information Theory*, pages 1748–1752, June 2010.
10. Frank Harary. The maximum connectivity of a graph. *Proceedings of the National Academy of Sciences of the United States of America*, 48(7):1142–1146, 1962.
11. F. Jaeger, D. L. Vertigan, and D. J. A. Welsh. On the computational complexity of the jones and tutte polynomials. *Mathematical Proceedings of the Cambridge Philosophical Society*, 108(1):35–53, 1990.
12. R. Jin, B. Wang, W. Wei, X. Zhang, X. Chen, Y. Bar-Shalom, and P. Willett. Detecting node failures in mobile wireless networks: A probabilistic approach. *IEEE Transactions on Mobile Computing*, 15(7):1647–1660, July 2016.
13. R. M. Karp. Reducibility among combinatorial problems. In R. E. Miller and J. W. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.
14. A. K. Kelmans. On graphs with randomly deleted edges. *Acta Mathematica Academiae Scientiarum Hungarica*, 37(1):77–88, 1981.
15. G. Kirchoff. Über die auflösung der gleichungen, auf welche man bei der untersuchung der linearen verteilung galvanischer ströme geführt wird. *Ann. Phys. Chem.*, 72:497–508, 1847.
16. Guillermo Rela, Franco Robledo, and Pablo Romero. Petersen graph is uniformly most-reliable. In *Machine Learning, Optimization, and Big Data*, pages 426–435, Cham, 2018. Springer International Publishing.
17. Mauricio G.C. Resende and Celso C. Ribeiro. *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*. Computational Science and Engineering. Springer-Verlag New York, 2016.
18. P. Romero. Building uniformly most-reliable networks by iterative augmentation. In *9th International Workshop on Resilient Networks Design and Modeling (RNDM)*, pages 1–7, 2017.
19. A. Rosenthal. Computing the reliability of complex networks. *SIAM Journal on Applied Mathematics*, 32(2):384–393, 1977.
20. A. Satyanarayana, L. Schoppmann, and C. L. Suffel. A reliability improving graph transformation with applications to network reliability. *Networks*, 22(2):209–216, 1992.
21. J Scott Provan and Michael O. Ball. The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM Journal on Computing*, 12(4):777–788, 11 1983.
22. Julián Viera. Búsqueda de grafos cúbicos de máxima confiabilidad. Master's thesis, Facultad de Ingeniería, Universidad de la República, Uruguay, 2018.
23. Guifang Wang. A proof of Boesch's conjecture. *Networks*, 24(5):277–284, 1994.

Chapter 6

Building Highly Reliable Networks with GRASP/VND Heuristics

In this chapter we continue growing with the number of node in (2,3)-graphs in order to find uniformly most-reliable graphs. An exhaustive computational test with cubic graphs with girth greater than 3 confirms that our resulting graphs achieve the maximum tree-number (therefore, they are the only candidates of uniformly most-reliable graphs), and the maximum girth as well. Finally we conjecture that regular uniformly most-reliable (p,q)-graph must have maximum girth.

Building Highly Reliable Networks with GRASP/VND Heuristics

Mathias Bourel, Eduardo Canale, Franco Robledo, Pablo Romero, Luis Stábile
Facultad de Ingeniería
Universidad de la República
Julio Herrera y Reissig 565, PC 11300
Montevideo, Uruguay
Emails: (mbourel,canale,frobledo,promero,lstabile)@fing.edu.uy

***Abstract*—There is a strong interplay between network reliability and connectivity theory. In fact, previous studies show that the graphs with maximum reliability, called uniformly most-reliable graphs, must have the highest connectivity.**

In this paper, we revisit the underlying theory in order to build uniformly most-reliable cubic graphs. The computational complexity of the problem promotes the development of heuristics.

The contributions of this paper are three-fold. In a first stage, we propose an ideal Variable Neighborhood Descent (VND) which returns the graph with maximum reliability. This VND works in exponential time. In a second stage, we propose a Greedy Randomized Adaptive Search Procedure (GRASP), that trades quality for computational effort. A construction phase enriched with a Restricted Candidate List (RCL) offers diversification. Our local search phase includes a globally optimum solution of an Integer Linear Programming (ILP) formulation. As a product of our research, we recovered previous optimal graphs from the related literature in the field. Additionally, we offer new candidates of uniformly most-reliable graphs with maximum connectivity and maximum number of spanning trees.

***Index Terms*—Network Optimization, Maximum Reliability, Heuristics, GRASP, VND, ILP.**

I. Motivation

In network reliability analysis, the goal is to find the probability of correct operation of a system [1], [2]. The context of the original problem determines our notion of correct operation. For instance, delay sensitive applications such as videoconference require a hop-constrained network, where the terminals should be connected by short paths [3]. Wireless systems deal with a hostile environment with mobility (fading, handover and coverage, among other challenges). The goal is to achieve a Grade of Service (GoS) during the busy hour, and node-reliability analysis is more suitable for this context [4]. The interaction between peers in a cooperative environment suggests potential links, and a link-reliability analysis is adequate for this context. Peer-to-peer systems suffer from starvation when the missing-piece syndrome affect all the system [5]. Clearly, the swarm (or population) should be connected, and the all-terminal reliability model is a suitable tool in order to understand this phenomena.

Several researchers from different fields of knowledge (mathematics, computer science, engineering), shaped the body of network reliability analysis, given the application and importance of the underlying models. A fundamental problem is to find the connectedness probability of a random graph, subject to link failures, called the *all-terminal reliability*. The scientific literature around this problem is vast; however, this problem is not fully understood yet. The corresponding practical problem is to connect p sites using q links in the *best* way, this is, to find the graph whose all-terminal reliability is maximum among all (p, q)-graphs. Such graphs are called *uniformly most-reliable graphs*.

The main contributions of this paper are the following:

1) An exact VND that returns uniformly most-reliable graphs is presented.
2) A hybrid GRASP/VND heuristic is introduced in order to find graphs with high reliability. It trades quality for computational feasibility.
3) An Integer Lineal Programming (ILP) formulation called Regularity Problem is proposed. The goal is to find a regular graph starting from a non-regular one moving as minimum number of links as possible.
4) A globally optimum solution for the Regularity Problem is proposed. It is included as a local search in our GRASP/VND solution.
5) Novel networks that show high reliability and connectivity are found, as a result of our hybrid heuristic.

The document is organized in the following manner. Section II formally states the problem and breakthroughs in the field of uniformly most-reliable graphs. Section III presents an exact VND that runs in exponential time, and a hybrid GRASP/VND heuristic that trades quality for computational feasibility. As a product, we offer novel cubic networks with high reliability in Section IV. Concluding remarks and open problems are discussed in Section V.

II. Uniformly Most-Reliable Graphs

A. Definition

In the following, we work with undirected graphs without loops, and a graph with p nodes and q links is a (p, q)-graph.

Definition 1. Consider a graph G with perfect nodes but independent link failures with identical probability $\rho \in (0, 1)$. The *all-terminal reliability*, $R_G(\rho)$, is the probability that the resulting subgraph remains connected.

The *unreliability* $U_G(\rho) = 1 - R_G(\rho)$ can be expressed using sum-rule:

$$U_G(\rho) = \sum_{k=0}^{q} m_k(G)\rho^k(1-\rho)^{q-k}, \tag{1}$$

$m_k(G)$ being the number of spanning disconnected subgraphs of G with exactly $q-k$ links. Therefore, $R_G(\rho)$ is a polynomial in $\rho \in (0, 1)$, and its determination is reduced to counting the numbers $\{m_k(G)\}_{k=0,\ldots,q}$.

Definition 2. A (p, q)-graph H is *uniformly most-reliable* if $R_H(\rho) \geq R_G(\rho)$ for all (p, q)-graph G and all $\rho \in (0, 1)$.

Alternatively, H is uniformly most-reliable if its unreliability $U_H(\rho)$ is dominated (i.e., upper-bounded) by all functions $U_G(\rho)$ for all (p, q)-graph G.

B. Breakthroughs

In this section we present fundamental results that are the cornerstone in the theory of uniformly most-reliable graphs. The following section briefly describes the main findings that complement the fundamental results.

In 1977, Arnie Rosenthal formally proved that the K-terminal reliability evaluation belongs to the class of $\mathcal{NP}$-Hard computational problems [6]. The key concept of the proof is the reducibility introduced in 1972 by Richard Karp, which represents a foundational work in computational complexity [7]. As corollary, finding uniformly most-reliable graphs is a hard problem as well.

Observe that if $m_k(H) \leq m_k(G)$ for all $k \in \{0, \ldots, q\}$ and (p, q)-graph G, then H is uniformly most-reliable. This is a simple but elegant interplay between network reliability analysis and connectivity theory. Curiously enough, the converse is still an open problem:

Conjecture 1 (Boesch et. al.)**.** If G is uniformly most-reliable (p, q)-graph, then $m_k(G) \leq m_k(H)$ for all (p, q)-graph H.

If $\lambda(H)$ denotes the connectivity of H and $\tau(H)$ its number of spanning trees, the following necessary criterion holds [8]:

Corollary 1. *A uniformly most-reliable graph H must have the maximum tree-number $\tau(H)$, maximum connectivity $\lambda(H)$, and the minimum number $m_\lambda(H)$.*

Corollary 1 wakes up interest in two special sub-problems: the maximum connectivity and maximum tree-number of a graph. In the second book ever written in graph theory, Claude Berge challenges the readers to find the graph with maximum connectivity among all graphs with a fixed number of nodes and links. Frank Harary provided not only a full answer, but also found connected graphs with minimum and maximum diameter [9]. The idea behind his construction is simple: by handshaking, the average degree of a (p, q)-graph is $\frac{2q}{p}$. Therefore, $\lambda \leq \lfloor \frac{2q}{p} \rfloor$. Harary graphs achieve this upper-bound, which represents the maximum connectivity of a graph.

Gustav Kirchhoff solved linear time-invariant resistive circuits, and as corollary he introduced the Matrix-Tree theorem, where he counts the number of spanning trees, or the tree-number, of a connected graph (i.e., the number $m_{q-p+1}(G)$), using the determinant of a matrix [10], [11]. This breakthrough in electrical systems launched the theory of trees, which provides the building blocks in communication design. However, the maximization of the tree-number among all (p, q)-graphs is not well understood.

For convenience we say that a (p, q)-graph, H, is t-optimal if $\tau(H) \geq \tau(G)$ for every (p, q) graph G. Briefly, Corollary 1 claims that uniformly most-reliable graphs must be t-optimal and max-λ min-m_λ, where λ denotes the link-connectivity.

To the best of our knowledge, the only general reliability improving graph transformation from the literature is called *swing surgery* by Satyanarayana et. al. [12], and it was discovered by Kelmans [13]. Specifically, if we are given a (p, q)-graph $G = (V, E)$, two nodes $x, y \in V$ with respective neighboring nodes S_x and S_y with $S_x - \{y\} \subset S_y$, $B \subseteq S_y - \{x\}$ such that $B \cap S_x = \emptyset$, and G' the graph obtained by G by removing the links $\{(y, z), z \in B\}$ and adding the links $\{(x, z), z \in B\}$, then $R_{G'}(\rho) \geq R_G(\rho)$ for all $\rho \in (0, 1)$.

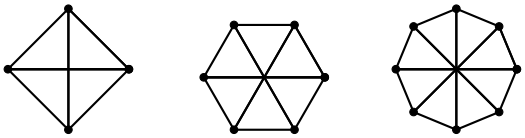
C. Findings

Redundancy is of paramount importance in communication networks. For this reason, in the following we are specifically focused on uniformly most-reliable cubic graphs (i.e., 3-regular connected graphs). By handshaking, this is the case of $(2r, 3r)$-graphs for some $r \geq 2$. Recall that Möbius graph M_n is precisely the elementary cycle C_{2n} together with all the diameters (opposite nodes are also linked). So far, the findings of uniformly most-reliable cubic graphs can be summarized in the following list:

- $K_4 = M_2$; complete graph with 4 nodes; case $r = 2$; see [14].
- $K_{(3,3)} = M_3$; complete bipartite graph; case $r = 3$; see [15].
- $W_4 = M_4$; Wagner graph ($r = 4$); see [16].
- P_5; Petersen graph ($r = 5$); see [17].
- Y_6; Yutsis graph ($r = 6$); see [18].

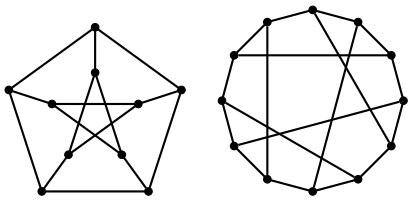
The reader is invited to consult the corresponding references for a formal proof that these graphs are uniformly most-reliable. They are sketched in Figures 1-2. By computational limits, currently it is not possible to find uniformly most-reliable cubic graphs by an exhaustive search

for $r \geq 7$. In previous works we built highly reliable cubic graphs for $r \geq 7, \ldots, 10$. The goal of this paper is to build highly reliable cubic graphs for $r \in \{11, \ldots, 15\}$, finding a trade-off between quality and computational effort.



(a) $K_4 = M_2$ (b) $K_{(3,3)} = M_3$ (c) M_4

Fig. 1. Complete, Bipartite and Wagner graphs.



(d) P_5 (e) Y_6

Fig. 2. Petersen and Yutsis graphs.

D. Equivalent Combinatorial Problem

It is worth remarking that the problem of finding uniformly most-reliable graphs is a simultaneous minimization of an uncountable family of numbers $\{U_G(\rho)\}_{\rho \in (0,1)}$. However, if Boesch Conjecture holds, we observe that the problem can be translated to a single-objective combinatorial optimization problem. Specifically, let us denote $m(G)$ as number of disconnected spanning subgraphs for G. By the definition of link disconnecting sets, we get that:

$$m(G) = \sum_{k=0}^{q} m_k(G).$$

The problem under study in this work is to minimize $m(G)$ restricted to the family of (p, q)-graphs, for p even and $q = 3p/2$.

Proposition 1. *Consider natural numbers p and q such that there exists a unique uniformly most-reliable (p, q)-graph. If Boesch conjecture holds, then G is uniformly most-reliable (p, q) graph if and only if $m(G)$ is minimum.*

Proof. Assume that G is uniformly most-reliable. By Boesch conjecture, every disconnecting set $m_k(G)$ is minimum among all the other (p, q)-graphs. Therefore, the number $m(G) = \sum_{k=0}^{q} m_k(G)$ is also minimum in this set. For the converse, consider a (p, q)-graph G such that $m(G)$ is minimum. By hypothesis, there exists a uniformly most-reliable (p, q)-graph, denoted by H. By Boesch conjecture, $m_k(H) \leq m_k(G)$. Since $m(G) = \sum_{k=0}^{q} m_k(G)$ is minimum, the only possibility is that $m_k(G) = m_k(H)$ for all k. Therefore, G and H share the same unreliability polynomial. By uniqueness, we must have $G = H$, and the statement is proved. □

In short, Proposition 1 tells us that if Boesch conjecture holds, then finding uniformly most-reliable graphs is equivalent to the minimization of disconnected spanning subgraphs $m(G)$. The optimum graphs under connectivity (i.e., purely deterministic) and reliability optics (probabilistic) share common properties.

Supported by Proposition 1, we are focused on the minimization of disconnected spanning subgraphs $m(G)$. In this paper, we offer highly reliable cubic graphs with strong connectivity properties as well.

It is well-known that finding the coefficients $\{m_k(G)\}_{k=0,\ldots,q}$ belongs to the hierarchy of #$\mathcal{P}$-Complete counting problems [19]. Furthermore, the number $m(G) = \sum_{k=0}^{q} m_k(G) = T(1, 2)$, is precisely Tutte polynomial evaluated at the point $(1, 2)$, which is a #$\mathcal{P}$-Hard counting problem, even for bipartite planar graphs [20]. Here, we propose a pointwise statistical estimation of this number. Monte Carlo is a noteworthy computational tool for simulation. From a macroscopic point of view, the idea is to faithfully simulate a complex system (or a part of it), and consider N independent experiments of that simulation, in order to determine the performance of the system (or subsystem) and assist decisions on it [21].

We will use Crude Monte Carlo (CMC) in order to provide an unbiased statistical estimation for $m(G)$. First of all, observe that $m(G)$ is strictly related with the unreliability evaluation at $\rho = \frac{1}{2}$:

$$U_G\left(\frac{1}{2}\right) = \sum_{k=0}^{q} m_k(G) \left(\frac{1}{2}\right)^k \left(\frac{1}{2}\right)^{q-k} = \frac{m(G)}{2^q}. \quad (2)$$

Equation (2) shows that, alternatively, we must minimize $U_G(\frac{1}{2})$. If we are given two graphs G and G', and we want to determine whether $m(G) < m(G')$ or not, we proceed as follows. First, we pick two independent random graphs $G_1, \ldots, G_n$ for G and $G'_1, \ldots, G'_n$ for G', with link failures $\rho = \frac{1}{2}$. Define independent Bernoulli variables $X_1, \ldots, X_n$ and $Y_1, \ldots, Y_n$ such that $X_i = 1$ if and only if G_i is disconnected (resp. $Y_i = 1$ if and only if G'_i is disconnected). Supported by the strong law of large numbers, we decide $m(G) < m(G')$ if and only if the respective averages $\bar{X}_n$ and $\bar{Y}_n$ verify the test $\bar{X}_n < \bar{Y}_n$. This criterion avoids the full determination of the coefficients $m_k(G)$, and it will be useful for the design of a GRASP/VND heuristic to build highly reliable graphs.

III. Metaheuristics

In this section we develop an ideal VND metaheuristic that returns a uniformly most-reliable graph. Since it has exponential time, we must trade accuracy for computational effort. As a consequence, a full GRASP/VND heuristic is introduced.

A. Exact VND

Variable Neighborhood Descent (VND) explores several neighborhood structures in a deterministic order. Its success is based on the simple fact that different neighborhood

structures do not usually have the same local minima. Thus, the local optima trap problem is addressed by a deterministic change of neighborhoods [22].

Recall that a simultaneous minimization of the coefficients $\{m_k(G)\}_{k=0,\dots,q}$ is a sufficient condition for G to be uniformly most-reliable. Therefore, if there is one local search dedicated to each coefficient, the output must be uniformly most-reliable. Trivial neighborhood structures where all (p,q)-graphs are neighbors of some fixed graph work. However, the cardinality of the search-space of (p,q)-graphs is $\binom{p(p-1)/2}{q}$. Therefore, an exhaustive search among the trivial neighborhood structures of all (p,q)-graphs is computationally prohibitive. This fact promotes the development of heuristics that trade quality for computational efficiency.

B. GRASP/VND Heuristic

GRASP is an iterative multi-start process which operates in two phases [23]. In the Construction Phase a feasible solution is built whose neighborhood is then explored in the Local Search Phase. The second phase is usually enriched by means of different variable neighborhood structures, for instance, VND.

We adapt the previous ideal VND in order to obtain a feasible computational solution in a multi-start fashion with diversification in a previous construction phase. Algorithm $HighlyReliable$ receives a maximum number of iterations $iter$, a natural number $r \geq 2$, and returns a highly reliable cubic $(2r, 3r)$-graph.

Algorithm 1 $G = HighlyReliable(r, iter, \alpha)$

```
1: G ← M_r
2: for i = 1 to iter do
3:    G_input ← Construct(r, α)
4:    G(i) ← VND(G_input)
5:    if m(G(i)) ≤ m(G) for all k then
6:       G ← G(i)
7:    end if
8: end for
9: return G
```

In Line 1, the graph G is initialized as Möbius graph M_r, which is known to be optimal for the cases where $r \in \{2,3,4\}$. In a **for**-loop with $iter$ iterations (Lines 2-8), we iteratively call in sequence the Construction Phase (Lines 3) and VND (Line 4). If the number of disconnected spanning subgraphs $m(G(i))$ is dominated by $m(G)$, the current graph G is replaced by $G(i)$ (Lines 5-6). It is worth to remark that the test $m(G(i)) \leq m(G)$ considers the test of averages detailed in Subsection II-D. The best graph among all the iterations is returned as the output (Line 9).

In the following, we provide details of the Construction Phase ($Construct(r,\alpha)$ from Line 3) and Local Search Phase (VND function, from Line 4). The result is not necessarily a uniformly most-reliable network, but a highly reliable cubic network, which is useful for practical purposes.

Algorithm 2 $G = Construct(r, \alpha)$

```
1: U ← RandomNumbers(r(r − 1)/2)
2: G ← RandomTree(U)
3: δ ← min_{v∈G}{deg(v)}
4: Δ ← max_{v∈G}{deg(v)}
5: RCL ← {(v_i, v_j) : deg(v_i) deg(v_j) ≤ δ^2 + α(Δ^2 − δ^2)}
6: for i = 1 to r + 1 do
7:    e_i ← Random(RCL)
8:    G ← G ∪ {e_i}
9:    RCL ← Update(RCL, e_i)
10: end for
11: return G
```

1) Construction Phase: The main idea is to start with a random tree with $2r-1$ links and insert adequately $r+1$ links meeting a final size of $3r$ links. In Line 1, for every pair of potential links we pick independent numbers in $(0,1)$ uniformly chosen at random. In this way, we get random costs c_{ij} for every pair of nodes v_i and v_j. A random tree is found in Line 2. Specifically, function $RandomTree$ applies Kruskal algorithm with the costs c_{ij}. The minimum and maximum degree of the resulting graph are found in Lines 3 and 4 respectively. The addition of the remaining $r+1$ links takes place in the block of Lines 5-10. A Restricted Candidate List, RCL, selects a percentage of α links $e_{(i,j)}$ with the lowest product degrees $\deg(v_i) \times \deg(v_j)$; see Line 5. In the **for**-loop of Lines 6-10, the remaining links are iteratively picked from the RCL (Line 7) and added to the graph G (Line 8). Observe that the RCL should be updated, since the degrees are modified in each iteration. This operation takes place in function $Update(RCL, e_i)$ (Line 9). Clearly, G is not necessarily regular, but the effect of the RCL provides diversity in the solutions. Naturally, it trades greediness for randomization with the parameter α, and tends to return almost-regular graphs.

2) Local Search Phase: In the Local Search Phase, a VND is considered with the following movements:

1) $Surgery$: applies the graph transformation called Swing Surgery (see Subsection II-B).
2) $Cubic$: returns a cubic graph after adequate link addition/deletions.
3) $Crossing$: tests whether the tree-number is increased after all feasible graph-crossings.

A graph G is *healthy* if there is no reliability-improving surgery. It is *strong* if G has the largest tree-number under all feasible crossings. Figure 3 presents the full VND diagram. The reader can observe that the output G is cubic, healthy and strong.

In the following, we explain the three movements. Let us start with $Surgery$ and $Crossing$ (as we will see, $Cubic$ includes a globally optimum solution of an ILP formulation). $Surgery$ just applies the reliability-improving graph transformation called Swing Surgery from Subsection II-B whenever possible. $Crossing$ tries to find a link-crossing with largest tree-number. Specifically, if the links $e_1 = (x,y)$

and $e_2 = (z, t)$ belong to G, but $e_3 = (x, z)$, $e_4 = (y, t)$ do not belong to G, $Crossing$ counts the tree-number of the new graph $G' = G - e_1 - e_2 + e_3 + e_4$. The tree-number is efficiently found by Kirchhoff theorem, as the magnitude of any cofactor of the Laplacian matrix [11].

The main idea of $Cubic$ is to return a cubic graph starting from a graph G such that $2|V| = 3|E|$. It returns the globally optimum solution to an Integer Linear Programming formulation. Consider the input graph $G = (V, E)$, the resulting cubic graph $G' = (V, E')$ and the following binary variables:

- $e_{(i,j)} = 1$ iff $(i, j) \in E$ (adjacency matrix for G);
- $a_{(i,j)} = 1$ iff $(i, j) \in E' - E$ (links added to G');
- $r_{(i,j)} = 1$ iff $(i, j) \in E - E'$ (links removed from G).

Our goal is to minimize the number of addition/deletions in order to return a cubic graph G'. The Regularity Problem can be formalized by the following ILP:

$$\min \sum_{i<j} a_{(i,j)} \tag{3}$$

$$s.t. \tag{4}$$

$$\sum_{i,j} a_{(i,j)} = \sum_{i<j} r_{(i,j)} \tag{5}$$

$$\sum_{j:j\neq i} a_{(i,j)} + e_{(i,j)} - r_{i,j} = 3 \, \forall i \in \{1, \ldots, 2r\} \tag{6}$$

$$a_{(i,j)}; r_{(i,j)} \in \{0, 1\} \, \forall i, j. \tag{7}$$

Constraint 5 states that the number of added/removed links must be identical, so, $|E| = |E'|$. Constraints 6 state that the resulting graph G' must be 3-regular. Finally, Constraints 7 determine the binary domain for the decision variables $a_{(i,j)}$ and $r_{(i,j)}$.

In the following, $Cubic$ movement is specified. The key idea is to move links from the set $E^+(G) = \{vw \in E : \deg_G(v), \deg_G(w) > 3\}$ into some adequate *potential link-subset* $F \subseteq E^-(G) = \{vw \notin E : \deg_G(v), \deg_G(w) < 3\}$. After the block of *translations* of type-I moves from Lines 1-6, we could have links $e = \{u, v\} \in E$ with degrees $\deg(u) < 3$ and $\deg(v) \geq 3$. This is solved by new *pivotal moves* or type-II moves, in a second block of Lines 7-11. In order to find an optimal F, consider $V_i = \{v \in V : \deg_G(v) = i\}$ and let $V^+ = \cup_{i>3} V_i$ the node-set with exceeding degrees.

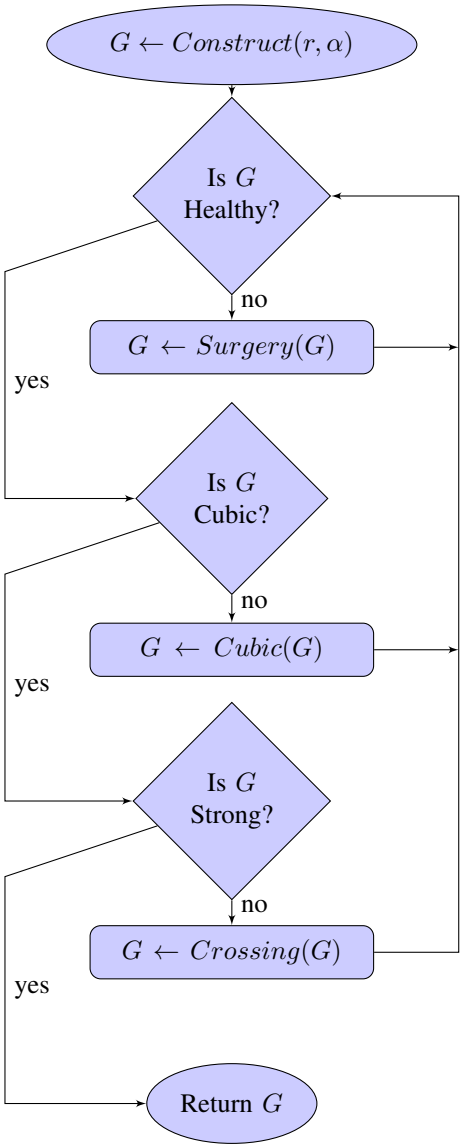


Fig. 3. Flow Diagram for the Local Search Phase - VND.

Algorithm 3 $G = Cubic(G)$

Require: $G = (V, E)$ connected with $3|V| = 2|E|$

1: $F \leftarrow PotentialLinks(G)$
2: **while** $|E^+(G)|, |F| > 0$ **do**
3: Get e from $E^+(G)$ and f from F
4: $G \leftarrow G - e + f$
5: $F \leftarrow F - f$
6: **end while**
7: **while** $\exists v \in V : \deg_G(v) > 3$ **do**
8: Find $v' \in V : \deg(v') < 3$
9: Find $w \in N_v - N_{v'}$
10: $G \leftarrow G - vw + v'w$
11: **end while**
12: **return** G

Similarly, let $V^- = V_1 \cup V_2$ be the node-set with deficient degrees. By Handshaking, the exceeding degree $\gamma(G)$ is equal to the total deficient degree, and

$$\gamma(G) = \sum_{x \in V^+} (\deg(x) - 3) = \sum_{x \in V^-} (3 - \deg(x)) = 2|V_1| + |V_2|. \tag{8}$$

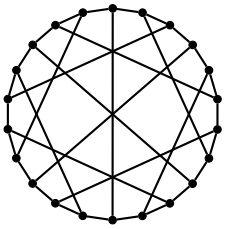
Since type-I moves are better than type-II moves in terms of the objective of the Regularity Problem, the best is to pick $|F|$ as large as possible. Indeed, the algorithm achieves the optimal value for the Regularity Problem if there exists some potential link-subset F such that $|F| \geq \gamma(G)/2 - 3/2$. We will construct such F in the proof of the following theorem. The existence of such subset in the general k-regular case is an open problem related with a generalization of the Erdös-Gallai theorem, but for cubic graph, it is an easy puzzle. Indeed, if $G' = (V^-, F)$, then $\deg_{G'}(x) \leq 2$ for all $x \in V_1$, while $\deg_{G'}(x) \leq 1$ for all $x \in V_1$. As a consequence, G' is a union of paths and cycles, which leads to a tractable problem.

Theorem. *For any graph G with $3|V| = 2|E|$ there is a set F of potential edges of size $\gamma(G)/2 - 3/2$ or more.*

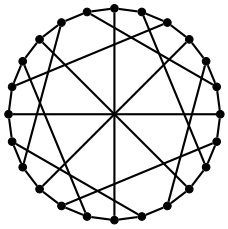
A full proof is included in the Appendix. As corollary, *Cubic* algorithm is globally optimum for the Regularity Problem, and it serves as one of the local searches in our GRASP/VND solution.

IV. Results

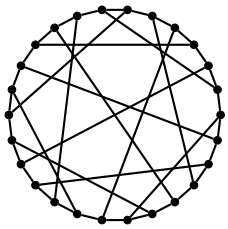
By construction, it is clear that our GRASP/VND heuristic returns M_r for the known cases $r \in \{2, 3, 4\}$ (see Line 1 of *Construction*). Figure 4 sketches the five resulting graphs for $r \in \{11, 12, 13, 14, 15\}$ using $m = 10^6$ iterations, $\alpha = 0.5$ and sample size $N = 10^4$ for Crude Monte Carlo for the pointwise reliability estimation of $U_G(\frac{1}{2})$ as detailed in Subsection II-D. All the graphs produced for $r = 11, \ldots, 15$ have the maximum tree-number among all cubic graphs. Figure 5 shows the evolution of the tree-number as a function of the number of iterations in the algorithm.



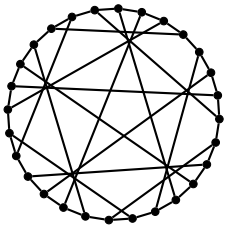
(a) $U_{R_{11}}(\frac{1}{2}) = 0.9801$
$\tau(R_{11}) = 32710656$



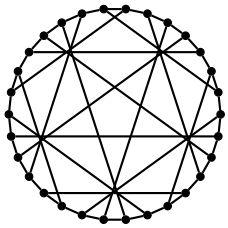
(b) $U_{R_{12}}(\frac{1}{2}) = 0.988$
$\tau(R_{12}) = 168664320$



(c) $U_{R_{13}}(\frac{1}{2}) = 0.9904$
$\tau(R_{13}) = 862488000$



(d) $U_{R_{14}}(\frac{1}{2}) = 0.9931$
$\tau(R_{14}) = 4410450000$



(e) $U_{R_{15}}(\frac{1}{2}) = 0.9956$
$\tau(R_{15}) = 23066015625$

Fig. 4. Results for $r \in \{11, \ldots, 15\}$

An exhaustive computational test with cubic graphs with girth greater than 3 confirms that our resulting graphs achieve the maximum tree-number (therefore, they are the only candidates of uniformly most-reliable graphs), and the maximum girth as well. We conjecture that this is the case for any uniformly most-reliable graph under regularity.

Conjecture 2. Regular uniformly most-reliable (p, q)-graph must have maximum girth.

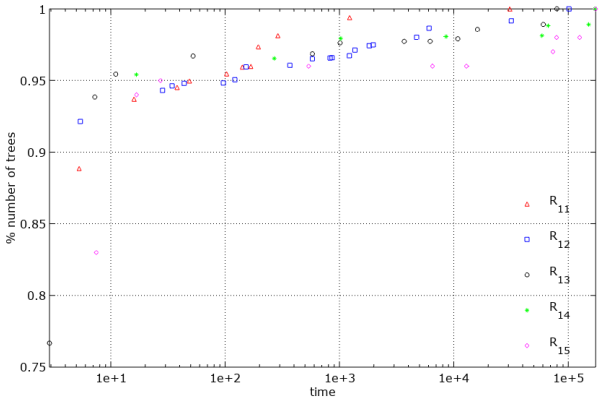


Fig. 5. Evolution of tree-number for $r \in \{11, \ldots, 15\}$.

V. Conclusions and Trends for Future Work

Reliability maximization is a relevant problem from network design. Potential applications include virtual and wireless systems, and cooperative environments in a hostile system, where the links may fail.

In the theory of reliability maximization, the goal is to find uniformly most-reliable graphs. The breakthroughs and main result of this theory are here outlined. It has half a century of development; however, there are key questions without concluding answers. Boesch conjecture and Wagner extension are just examples.

Supported by the computational complexity of the problem, we first present a VND metaheuristic that returns all uniformly most-reliable graphs. The main drawback of this proposal is the computational effort. As a consequence, we then develop a hybrid GRASP/VND heuristic that keeps the most meaningful elements of the first VND implementation, enriched with a construction phase in order to gain diversification.

The first results are encouraging. Our hybrid GRASP/VND was able to return new candidates of such optimal graphs from a reliability viewpoint.

There are several trends for future work. A powerful methodology to find uniformly most-reliable graphs is not known. The power of new reliability-improving graph transformations is not considered yet.

APPENDIX

Theorem. *For any graph G with $3|VG| = 2|EG|$ the largest possible set F of potential edges has size $\gamma(G)/2 - 3/2$ or more.*

Proof. We consider the induced subgraphs $G_1 = G(V_1)$ and $G_2 = G(V_2)$ by the nodes V_1 and V_2 of degrees 1 and 2, respectively. Since G is connected, the only possibility is that G_1 has isolated nodes, and G isolated paths (it cannot have cycles since it would represent a whole isolated component from G).

We construct F as a function of $k = |V_1|$ and $h = |V_2|$. We can sort the nodes $V_1 = \{u_1, \ldots, u_k\}$ and $V_2 = \{v_1, \ldots, v_h\}$ in such a way that if $(v_i, v_j) \in E(G_2)$, then $|i - j| = 1$, since they are isolated paths. In order to define F we need some previous definitions: if $h, k \geq 3$, let $F_1 = \{u_1u_2, \ldots, u_ku_1\}$ and let

$$F_2(v_1, \ldots, v_l) = \begin{cases} \emptyset & l \leq 2, \\ \{v_1v_3, v_2v_4\} & l = 4, \\ \{v_1v_l\} \cup F_2(v_2, \ldots, v_{l-1}) & \text{otherwise.} \end{cases}$$

Finally, let $F_3(w_1, w_2, \ldots, w_l)$ be link-set defined recursively as:

$$F_3(w_1, w_2, \ldots, w_l) = \begin{cases} \{u_1u_2, w_1w_2\} & l = 2, I \neq \emptyset, \\ \{u_1u_2, u_1w_i, u_2w_{3-i}\} & l = 2, I = \emptyset, u_1 \not\sim w_i, \\ F_3(w_i, w_j) & l = 3, i < j, \{i, j\} \cap I = \emptyset, \\ F_3(w_i, w_{i+2}) \cup \{w_{3-i}w_{5-i}\} & l = 4, \{i, i+2\} \cap I = \emptyset \\ F_3(w_2, \ldots, w_{h-1}) \cup \{w_1w_h\} & l \geq 5, \end{cases}$$

where $I = \{i : u_1 \sim w_i \sim u_2\}$.

Then, we define F as follows, where $s = (h \mod 2)/2$:

l	h	F	$\frac{1}{2}\gamma(G) - \lvert F\rvert$
0	≥ 3	$F_2(v_1, \ldots, v_l)$	s
≥ 3	0	F_1	0
≥ 3	≥ 3	$F_1 \cup F_2(v_1, \ldots, v_l)$	s
1	0	$\emptyset$	1
1	1	$\{u_1v_1\}$ if $u_1 \not\sim v_1$	1/2
1	1	$\emptyset$ if $u_1 \sim v_1$	3/2
1	2	$\{u_1v_1, u_1v_2\}$ if $v_1 \not\sim u_1 \not\sim v_2$	0
1	2	$\{u_1v_i\}$ if $u_1 \sim v_{3-i}$	1
1	3	$\{u_1v_i, u_1v_j\}$ if $v_i \not\sim u_1 \not\sim v_j$	1/2
1	≥ 4	$\{u_1v_1, u_1v_h\} \cup F_2(v_2, \ldots, v_{h-1})$	s
2	0	$\{u_1u_2\}$	1
2	1	$\{u_1u_2\}$ if $u_1 \sim v_1 \sim u_2$	3/2
2	1	$\{u_1u_2, u_iv_1\}$ if $u_i \not\sim v_1$	1/2
2	2	$F_3(v_1, v_2)$	$1_{I \neq \emptyset}$
2	≥ 3	$F_3(v_1, \ldots, v_h)$	s

TABLE I
DEFINITION OF F.

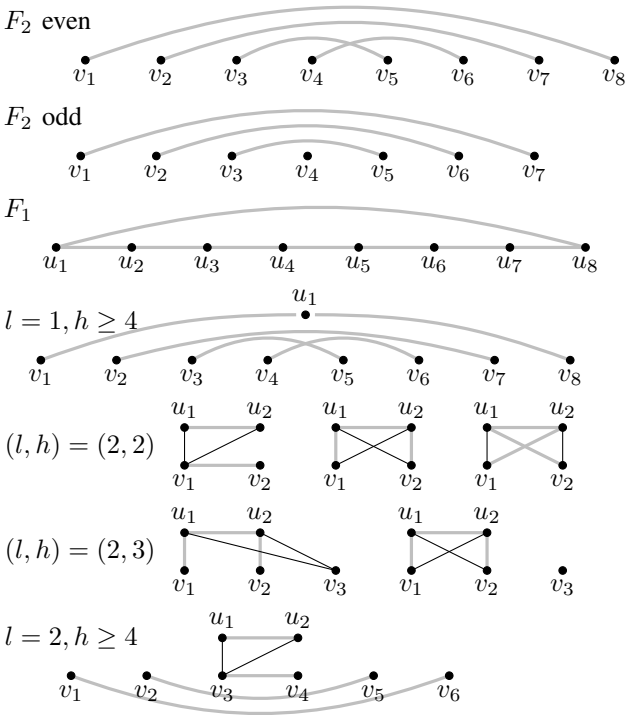


Fig. 6. Examples of F in gray. In black edges of G.

As the last column shows, the statement is verified in all the cases and it is clear that it cannot be improved. □

ACKNOWLEDGMENT

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos*.

REFERENCES

[1] C. J. Colbourn, "Reliability issues in telecommunications network planning," in *Telecommunications network planning, chapter 9*. Kluwer Academic Publishers, 1999, pp. 135–146.

[2] L. Beineke, R. Wilson, and O. Oellermann, *Topics in Structural Graph Theory*, ser. Encyclopedia of Mathematics and its Applications. Cambridge University Press, 2012.

[3] E. Canale, H. Cancela, F. Robledo, P. Romero, and P. Sartor, "Full complexity analysis of the diameter-constrained reliability," *International Transactions in Operational Research*, vol. 22, no. 5, pp. 811–821, 2015.

[4] R. Jin, B. Wang, W. Wei, X. Zhang, X. Chen, Y. Bar-Shalom, and P. Willett, "Detecting node failures in mobile wireless networks: A probabilistic approach," *IEEE Transactions on Mobile Computing*, vol. 15, no. 7, pp. 1647–1660, July 2016.

[5] B. Hajek and J. Zhu, "The missing piece syndrome in peer-to-peer communication," in *2010 IEEE International Symposium on Information Theory*, June 2010, pp. 1748–1752.

[6] A. Rosenthal, "Computing the reliability of complex networks," *SIAM Journal on Applied Mathematics*, vol. 32, no. 2, pp. 384–393, 1977.

[7] R. M. Karp, "Reducibility among combinatorial problems," in *Complexity of Computer Computations*, R. E. Miller and J. W. Thatcher, Eds. Plenum Press, 1972, pp. 85–103.

[8] D. Bauer, F. Boesch, C. Suffel, and R. V. Slyke, "On the validity of a reduction of reliable network design to a graph extremal problem," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 12, pp. 1579–1581, 1987.

[9] F. Harary, "The maximum connectivity of a graph," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 48, no. 7, pp. 1142–1146, 1962.

[10] G. Kirchoff, "Über die auflösung der gleichungen, auf welche man bei der untersuchung der linearen verteilung galvanischer ströme geführt wird," *Ann. Phys. Chem.*, vol. 72, pp. 497–508, 1847.

[11] N. Biggs, *Algebraic Graph Theory*, ser. Cambridge Mathematical Library. Cambridge University Press, 1993.

[12] A. Satyanarayana, L. Schoppmann, and C. L. Suffel, "A reliability improving graph transformation with applications to network reliability," *Networks*, vol. 22, no. 2, pp. 209–216, 1992.

[13] A. K. Kelmans, "On graphs with randomly deleted edges," *Acta Mathematica Academiae Scientiarum Hungarica*, vol. 37, no. 1, pp. 77–88, 1981.

[14] F. T. Boesch, X. Li, and C. Suffel, "On the existence of uniformly optimally reliable networks," *Networks*, vol. 21, no. 2, pp. 181–194, 1991.

[15] G. Wang, "A proof of Boesch's conjecture," *Networks*, vol. 24, no. 5, pp. 277–284, 1994.

[16] P. Romero, "Building uniformly most-reliable networks by iterative augmentation," in *9th International Workshop on Resilient Networks Design and Modeling (RNDM)*, 2017, pp. 1–7.

[17] G. Rela, F. Robledo, and P. Romero, "Petersen graph is uniformly most-reliable," in *Machine Learning, Optimization, and Big Data*. Cham: Springer International Publishing, 2018, pp. 426–435.

[18] J. Viera, "Búsqueda de grafos cúbicos de máxima confiabilidad," Master's thesis, Facultad de Ingeniería, Universidad de la República, Uruguay, 2018.

[19] J. Scott Provan and M. O. Ball, "The complexity of counting cuts and of computing the probability that a graph is connected," *SIAM Journal on Computing*, vol. 12, no. 4, pp. 777–788, 11 1983.

[20] F. Jaeger, D. L. Vertigan, and D. J. A. Welsh, "On the computational complexity of the jones and tutte polynomials," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 108, no. 1, p. 35?53, 1990.

[21] G. S. Fishman, *Monte Carlo: Concepts, Algorithms and Applications*. New York, NY, USA: Springer Verlag, 1996.

[22] A. Duarte, N. Mladenović, J. Sánchez-Oro, and R. Todosijević, *Variable Neighborhood Descent*. Cham: Springer International Publishing, 2016, pp. 1–27.

[23] M. G. Resende and C. C. Ribeiro, *Optimization by GRASP - Greedy Randomized Adaptive Search Procedures*, ser. Computational Science and Engineering. Springer-Verlag New York, 2016.

Part III

System Reliability

Chapter 7

Analysis and Reliability of Separable Systems

A Stochastic Binary System is a mathematical model of multi-component on-off system, where its components are subject to random failures. Considering the number of feasible states is computationally prohibitive. In this chapter we present a special class of SBS called *separable system*, that accept an efficient representation. Besides we proved that, even in this case, their reliability evaluation is computationally hard.

ORIGINAL ARTICLE

Networks

Analysis and Reliability of Separable Systems

Héctor Cancela[1]* | Graciela Ferreira[2]* | Gustavo Guerberoff[2]* | Franco Robledo[1,2]* | Pablo Romero[1,2]* | Luis Stábile[1,2]*

[1]Departamento de Investigación Operativa, Facultad de Ingeniería, Montevideo, Montevideo, 11300, Uruguay

[2]Laboratorio de Probabilidad y Estadística, Facultad de Ingeniería, Montevideo, Montevideo, 11300, Uruguay

Correspondence
Pablo Romero, Laboratorio de Probabilidad y Estadística, Facultad de Ingeniería, Montevideo, Montevideo, 11300, Uruguay
Email: promero@fing.edu.uy

Present address
† Laboratorio de Probabilidad y Estadística, Facultad de Ingeniería, Montevideo, Montevideo, 11300, Uruguay

Funding information
CSIC, Sistemas Binarios Estocásticos Dinámicos, Grant 395
Project MATHAMSUD 19-MATH-03

A Stochastic Binary System (SBS) is a mathematical model of multi-component on-off systems subject to random failures. An SBS models complex interactions between the states of the individual components and the operation of a global system. The reliability evaluation of SBSs is $\mathcal{NP}$-Hard, since it subsumes the classical network reliability evaluation. Furthermore, the number of states is exponential with respect to the number of components of the system. As a consequence, the representation of an SBS becomes a key element in order to develop exact or approximation methods for reliability evaluation.

The contributions of this article are multi-fold. First, we present the concept of *separable* systems, a special subset of SBSs that accept a compact representation. Second, we provide reliability bounds for arbitrary SBSs inspired by separable systems. Third, we formally prove that the reliability evaluation of separable systems is still $\mathcal{NP}$-Hard. However, we fully characterize separable systems under the all-terminal reliability model, finding that they admit efficient reliability evaluation in this relevant context.

KEYWORDS
Stochastic Binary System, Network Reliability, Computational

Abbreviations: SBS(s), Stochastic Binary System(s); SMBS(s), Stochastic Monotone Binary System(s).

* Equally contributing authors.

Complexity, Separable systems, Reliability bounds, Mathematical Programming

1 | INTRODUCTION

In system reliability analysis, the goal is to find the probability of correct operation of a system subject to component failures. A common practical problem is to design a system with maximum reliability meeting budget constraints [1, 2].

Classical network reliability analysis shaped the body of this field. In this basic setting, we are given a connected graph G with perfect nodes, and the links work independently with identical probability r. The *all-terminal reliability*, $R_G(r)$, is the probability that the resulting subgraph remains connected. This model and some variants (such as perfect links and nodes subject to failure) has been employed to model reliability of classical communications networks, where the emphasis was on a fixed infrastructure of sites holding communication equipment and of fixed links connecting them. Nevertheless, these models have limitations to represent the more diverse landscape of communication networks infrastructure, relying on different equipments, paradigms, and particularly in the case of wireless networks, where usually there does not exist a fixed, predetermined topology.

Stochastic binary systems (SBS) generalize the static reliability concept to any system composed of a number of components subject to independent failures with known probabilities, and where the operation or failure of the system as a whole is a function of the state of the individual components. In this sense, SBS are a more flexible tool for evaluating and optimizing the reliability of a wider spectrum of real systems, both in the networking area and in other quite different applications area [3, 4, 5]. At the same time, SBS present their own challenges in terms of computational analysis, as the evaluation of the reliability a general stochastic binary system belongs to the class of $\mathcal{NP}$-Hard problems. This has motivated different research efforts, tackling efficient exact methods for some subclasses of SBS, as well as approximations for the general case [6, 7, 8, 9, 10].

The contributions of this article can be summarized in the following items:

1. The concept of separable systems is presented.
2. Reliability bounds for arbitrary SBSs are provided, inspired by separability, duality and Chernoff inequality.
3. The $\mathcal{NP}$-Hardness of the reliability evaluation for separable systems is established.
4. A separable system under the all-terminal reliability model is called a separable graph. We fully characterize separable graphs. As a corollary, we conclude that the reliability evaluation of separable graphs can be obtained in linear time.
5. A discussion of the level of separability for non-separable systems is offered.

It is worth remarking that the concept of separable system was recently introduced in [11] by the same authors of the present paper. The reliability bounds were also presented in that previous conference work. Here, an extended analysis includes the hardness of the reliability evaluation of separable systems, as well as its application to the celebrated all-terminal reliability model, and a discussion of open problems.

The document is organized as follows. Section 2 presents fundamental concepts and examples of stochastic binary systems. Separable systems are presented in Section 3. Reliability bounds for arbitrary SBS are found in Section 4 using separable systems.

A theoretical analysis of separable systems is covered in Section 5. It includes a full characterization of separable systems in terms of Functional Analysis, as well as the hardness of the reliability evaluation for these systems. A

particular analysis of the all-terminal reliability model is offered in Section 6. Finally, Section 7 presents a preliminary analysis of nonseparable systems, concluding remarks and trends for future work.

2 | STOCHASTIC BINARY SYSTEMS

The following terminology is adapted from [12].

Stochastic Binary System A *stochastic binary system* is a triad (S, r, ϕ):

- $S = \{1, \ldots, N\}$ is a ground set of *components*,
- $r = (r_1, \ldots, r_N)$ are their *elementary reliabilities*, and
- $\phi : \{0,1\}^N \to \{0,1\}$ is the *structure*.

The concept of reliability is generalized to arbitrary stochastic binary systems.

Reliability/Unreliability Let $\mathcal{S} = (S, p, \phi)$ be a stochastic binary system, and consider a random vector $X = (X_1, \ldots, X_N)$ with independent coordinates governed by Bernoulli random variables such that $P(X_i = 1) = r_i$. The *reliability* of $\mathcal{S}$ is the probability of correct operation of the system:

$$R_{\mathcal{S}} = P(\phi(X) = 1) = E(\phi(X)) = \sum_{x:\phi(x)=1} P(X = x). \tag{1}$$

The *unreliability* of $\mathcal{S}$ is $U_{\mathcal{S}} = 1 - R_{\mathcal{S}}$.

A stochastic binary system is *homogeneous* if the elementary reliabilities are identical (i.e., $r_i = r$ for all i). In this paper we deal with homogeneous SBSs.

Pathsets/Cutsets Let $\mathcal{S} = (S, r, \phi)$ be a stochastic binary system. A possible state or configuration $x \in \{0,1\}^N$ is a *pathset* (resp. *cutset*) if $\phi(x) = 1$ (resp., if $\phi(x) = 0$).

The binary set $\{0,1\}$ is equipped with the partial order, defined by $0 \le 0, 0 \le 1$ and $1 \le 1$. The set $\{0,1\}^N$ inherits a natural order in the Cartesian product. Given two partially ordered sets A and B, a function $f : A \to B$ is monotonically increasing if $f(a_1) \le f(a_2)$ whenever $a_1 \le a_2$. As usual, we denote $y < x$ if $y \le x$ and $y \ne x$. Let us denote by $\vec{0}_N$ (resp. $\vec{1}_N$) the binary word with all bits set to 0 (resp. to 1), and by δ_i the binary word with all bits in 0 except the bit in position i which is set to 1.

Stochastic Monotone Binary System (SMBS) The triad $\mathcal{S} = (S, r, \phi)$ is a *stochastic monotone binary system* if the structure function $\phi : \{0,1\}^N \to \{0,1\}$ is monotonically increasing, $\phi(\vec{0}_N) = 0$ and $\phi(\vec{1}_N) = 1$.

Observe that SMBSs represent *well-behaved* SBSs, in the sense that, given a working configuration, the system can fail after the removal of some components, but can not fail if some failed components start to work. Additionally, the system does not work if it has no operational components, and the full system works.

Minpaths/Mincuts/Rays Let $\mathcal{S} = (S, r, \phi)$ be an SMBS:

- A pathset x is a *minpath* if $\phi(y) = 0$ for all $y < x$.

- A cutset y is a *mincut* if $\phi(x) = 1$ for all $x > y$.
- The x-ray is the set $S_x = \{y \in \{0,1\}^N : y \geq x\}$.

An SMBS is fully characterized by its mincuts (or its minpaths). In fact, if we are given the complete list of minpaths, then the complete list of pathsets is precisely the union of the x-rays over all minpaths x.

We will denote by $\overline{x}$ the state complementary to x in bits (i.e., 0 in x are set to 1 in $\overline{x}$, and vice-versa). In particular, $\overline{\phi(x)} = 1 - \phi(x)$. The following definition of duality will be useful for our later analysis of monotonicity and bounds [13]:

Duality The dual of a stochastic binary system $\mathcal{S} = (S, r, \phi)$ has identical ground set S, elementary reliabilities $r_i^d = 1 - r_i$, and structure $\phi^d(x) = 1 - \phi(\overline{x})$, for all possible states $x \in \{0,1\}^N$. The dual is denoted by $\mathcal{S}^d = (S, 1 - r, \phi^d)$.

The following examples provide an insight of the different applications of stochastic binary systems. Classical examples include a reference in the field for the interested reader.

1. All-Terminal Reliability: the ground set is the set of links of a simple graph. The system is up if the resulting random graph is connected.
2. K-Terminal Reliability: in the same random graph, the system is up if some distinguished node-set K, called the terminals set, is included in a connected component [14].
3. Source-Terminal Reliability: the previous model with $K = \{s, t\}$.
4. Diameter Constrained Reliability: a diameter constraint d is added to the K-Terminal Reliability. The system is up if every pair of terminals is connected by paths whose length is not greater than d [15, 16].
5. Node-Reliability: the ground set is the set of the nodes of a simple graph. The system is up if the resulting random graph is connected [17].
6. Forbidden patterns: given a binary word w, the operation rule is that $\phi_w(x) = 1$ if and only if w appears as a sub-word of x. This example shows that an SBS is not always represented by a network.

The reader can observe that all the examples are SMBSs except for the last one, which is not presented in terms of a network.

There exists a direct connection between SBSs and propositional logic. Recall that a theorem-proving procedure is the first $\mathcal{NP}$-Complete decision problem established by Stephen Cook [18]. In other words, the recognition of a tautology is a hard decision problem from propositional logic.

Theorem 1 *The reliability evaluation of an arbitrary SMBS belongs to the class of $\mathcal{NP}$-Hard problems.*

Proof Arnie Rosenthal proved that the reliability evaluation for the K-terminal reliability model belongs to the class of $\mathcal{NP}$-Hard computational problems [19]. Since K-Terminal is a particular SMBS, the result follows by inclusion.

Corollary 2 *The reliability evaluation of an arbitrary SBS belongs to the class of $\mathcal{NP}$-Hard problems.*

Les us close this section with three elementary properties of the dual system that will be useful in our analysis.

Lemma 3 *The dual of the dual is the original system.*

Proof $\phi^{dd}(x) = 1 - \phi^d(\overline{x}) = 1 - (1 - \phi(x)) = \phi(x)$.

Lemma 4 *The dual of an SMBS is another SMBS.*

Proof Consider arbitrary states $x \leq y$ and a monotone structure ϕ. Since $\overline{x} \geq \overline{y}$, we get that $\phi(\overline{y}) \leq \phi(\overline{x})$. Therefore: $\phi^d(x) = 1 - \phi(\overline{x}) \leq 1 - \phi(\overline{y}) = \phi^d(y)$.

The dual system has complementary reliability with respect to the original one:

Lemma 5 *If $\mathcal{S} = (S, \phi, p)$ is an SBS, then $R_{\mathcal{S}^d} = 1 - R_{\mathcal{S}}$.*

Proof Recall that the dual system has complementary probabilities in every component. Therefore: $P^d(X = x) = \prod_{i:x_i=1}(1 - r_i) \prod_{i:x_i=1}(r_i) = P(X = \overline{x})$. Let $\mathcal{P}$ denote the path-sets of the original SBS. Then:

$$\begin{aligned} R_{\mathcal{S}^d} &= \sum_{x:\phi^d(x)=1} P^d(X = x) = \sum_{x:\phi(\overline{x})=0} P^d(X = x) \\ &= 1 - \sum_{x:\phi(\overline{x})=1} P^d(X = x) \\ &= 1 - \sum_{x:\phi(\overline{x})=1} P(X = \overline{x}) \\ &= 1 - P(x \in \mathcal{P}) = 1 - R_{\mathcal{S}}. \end{aligned}$$

3 | SEPARABLE SYSTEMS

Observe that $\{0,1\}^N$ is the set of the extremal points of the unit hypercube $Q_N \subseteq \mathbb{R}^N$. Let us assign labels to the extremal points of Q_N according to a given structure ϕ. Every hyperplane defines a partition of $\mathbb{R}^N$ into two subsets. Consider the family of hyperplanes $\mathcal{H}$ such that $\vec{0}_N$ and $\vec{1}_N$ lie on different sides. For any member H of $\mathcal{H}$, denote by $Q_0 \subseteq Q_N$ the extremal points of the hypercube that belong to the side of $\vec{0}_N$; and $Q_1 = Q_N - Q_0$. Define a structure function ϕ_H such that its cutsets are precisely Q_0, and its pathsets are Q_1. Consider an equivalence relation $(\mathcal{H}, \sim)$ such that $H_1 \sim H_2$ if and only if $\phi_{H_1} = \phi_{H_2}$.

Recall that in the Euclidean space $\mathbb{R}^N$, a hyperplane is fully characterized by a normal vector $\vec{n}$ and a point P that belongs to the hyperplane: $\langle \vec{n}, X - P \rangle = 0$, where $\langle x, y \rangle = \sum_{i=1}^N x_i y_i$ is the inner product. If we denote $\vec{n} = (n_1, \ldots, n_N)$ and $\langle \vec{n}, P \rangle = \alpha_0$, the hyperplane can be written as $\sum_{i=1}^N n_i x_i = \alpha_0$. By convention and without loss of generality, we will consider that cutsets lie on the hyperplane or in its negative side, so that they verify $\sum_{i=1}^N n_i x_i \leq \alpha_0$, and that pathsets lie on the positive side of the hyperplane, and verify $\sum_{i=1}^N n_i x_i > \alpha_0$.

Lemma 6 *Consider a monotone structure ϕ. If $\phi = \phi_H$ for some hyperplane H, then there exists $H' \sim H$ with non-negative normal vector such that $\|\vec{n}\|_1 = \sum_{i=1}^N n_i = 1$.*

Proof Let $\phi = \phi_H$ for the hyperplane H) $\sum_{i=1}^N n_i x_i = \alpha_0$, and suppose that there exists some index j such that $n_j < 0$. There are two exhaustive and mutually disjoint cases:

- **i** There exists some mincut $x = (x_1, \ldots, x_N)$ such that $x_j = 0$: in this case, we know that $x + \delta_j$ is a minpath, so, $\phi(x + \delta_j) = 1$. By the definition of the separating hyperplane and the structure function ϕ_H, we get that $\sum_{i=1}^N n_i x_i \leq \alpha_0$ but $\sum_{i=1}^N n_i x_i + n_j > \alpha_0$. The only possibility is that $n_j > 0$. But we assumed $n_j < 0$; this is a contradiction.
- **ii** All mincuts verify $x_j = 1$: Consider an alternative hyperplane H') $\sum_{i \neq j}^N n_i x_i = \alpha_0 - n_j$. We will prove that $H' \sim H$. If x is a mincut, then $\sum_{i=1}^N n_i x_i \leq \alpha_0$, and therefore $\sum_{i \neq j}^N n_i x_i \leq \alpha_0 - n_j$. If x is a minpath, it must have $x_j = 1$. Since

$\sum_{i=1}^{N} n_i x_i > \alpha_0$ we get that $\sum_{i\neq j}^{N} n_i x_i > \alpha_0 - n_j$. Observe that $n_j = 0$ in the new hyperplane H', and $H' \sim H$ as desired.

By an iterative replacement of all the negative coordinates we obtain an equivalent hyperplane $H' \sim H$ with non-negative vector $\vec{n'}$, expressed by $H') \sum_{i=1}^{N} n'_i x_i = \alpha'$ for some real number α'. Finally, observe that $\vec{0}_N$ is always a cutset, so $0 \leq \alpha'$. Analogously, $\vec{1}_N$ is always a pathset, so $\sum_{i=1}^{N} n'_i > \alpha' \geq 0$. The result is obtained by a normalization of the resulting vector, which is possible since $\sum_{i=1}^{N} n'_i > 0$.

Even though there exist infinite equivalent hyperplanes, using Support Vector Machine (SVM) it is possible to find a single hyperplane with the largest gap (this is, with the largest distance to any of the vertices in the hypercube). Using Lemma 6, we can replace it by an equivalent hyperplane with non-negative versor. Without loss of generality, we will assume a non-negative normal vector with unit 1-norm.

Proposition 7 *The structures ϕ_H are monotone.*

Proof By Lemma 6, in particular we can choose $n_i \geq 0$ in the hyperplane $H) \sum_{i=1}^{N} n_i x_i = \alpha_0$. Let us denote $f(x) = \sum_{i=1}^{N} n_i x_i$. If $x \leq y$, then $f(x) \leq f(y)$, and therefore $\phi_H(x) \leq \phi_H(y)$.

A subtlety is that the *mincuts* from Lemma 6 are indeed the points $Q_0 \subset Q_N$ that are closer to the original hyperplane. A natural question is to determine if all SMBSs can be represented by a hyperplane. The answer is negative:

Proposition 8 *There exist SMBSs that cannot be represented by a hyperplane.*

Proof Consider the SMBS defined by the mincuts $M = \{(1,1,0,0),(0,0,1,1)\}$. Then the set of minpaths P is defined by $P = \{(1,0,1,0),(1,0,0,1),(0,1,1,0),(0,1,0,1),\}$. Suppose for a moment that there exists some separator $H) \sum_{i=1}^{4} n_i x_i = \alpha$ for some real numbers $\alpha, n_1, \ldots, n_4$. Since $(1,1,0,0)$ and $(0,0,1,1)$ are mincuts, we get that $n_1 + n_2 \leq \alpha$ and $n_3 + n_4 \leq \alpha$, so that $\sum_{i=1}^{4} n_i \leq 2\alpha$. However, if we consider the two minpaths $(1,0,1,0)$ and $(0,1,0,1)$, it holds that $n_1 + n_3 > \alpha$ and $n_2 + n_4 > \alpha$, so that $\sum_{i=1}^{4} n_i > 2\alpha$; a contradiction.

Separable System An SBS is *separable* if the cutsets/pathsets can be separated by some hyperplane.

An interpretation of separable systems recalls Riesz representation theorem for Hilbert spaces [20]. Indeed, the structure of a separable system can be written as an indicator that an inner-product exceeds some threshold in a Hilbert space:

$$\phi(x) = 1_{\langle x, \vec{n} \rangle \geq \alpha_0}. \tag{2}$$

A separable system can be represented with $N+1$ real numbers, instead of a logic table of 2^N values for an arbitrary SBS. This compact, space-efficient representation is a key point of our interest in separable systems.

4 | RELIABILITY BOUNDS

In this section we exploit the properties shared by separable systems in order to find reliability bounds for arbitrary SBSs. The strategy is the following:

- First, we find an upper bound for the reliability of separable systems using Chernoff inequality [21].
- For any given structure ϕ, we find the closest separable systems $\overline{\phi}$ and $\underline{\phi}$ such that $\underline{\phi} \leq \phi \leq \overline{\phi}$.
- By means of Chernoff-upper bound for separable systems and $\overline{\phi}$, we produce an upper bound for the reliability of the original SBS.
- By means of Chernoff-upper bound for separable systems, $\underline{\phi}$ and duality, we produce a lower bound for the reliability of the original SBS.

We describe each step in the following subsections.

4.1 | Chernoff Bound

Lemma 9 *For all separable systems $\mathcal{S}$ we have:*

$$R_{\mathcal{S}} \leq e^{-\sup_{t>0}\left\{t\alpha_0 - \sum_{i=1}^{N} c(i,t)\right\}}, \tag{3}$$

being $c(i,t) = \log \mathbb{E}\left(e^{t n_i x_i}\right) = \log(p.e^{t n_i} + 1 - p)$.

Proof We apply a well-known result introduced by Chernoff and used in the proof of Cramér Theorem for large deviations bounds [21]. For each $t > 0$ we have:

$$\begin{aligned} R_{\mathcal{S}} &= \mathbb{P}(\phi(x) = 1) = \mathbb{P}(\sum_{i=1}^{N} n_i x_i \geq \alpha_0) \\ &= \mathbb{P}(t\sum_{i=1}^{N} n_i x_i \geq t\alpha_0) = \mathbb{P}(e^{t\sum_{i=1}^{N} n_i x_i} \geq e^{t\alpha_0}) \\ &\leq e^{-t\alpha_0}\mathbb{E}\left(e^{t\sum_{i=1}^{N} n_i x_i}\right) = e^{-t\alpha_0}\prod_{i=1}^{N}\mathbb{E}\left(e^{t n_i x_i}\right) \\ &= e^{-t\alpha_0}\prod_{i=1}^{N}\left(e^{c(i,t)}\right) = e^{-\{t\alpha_0 - \sum_{i=1}^{N} c(i,t)\}}, \end{aligned} \tag{4}$$

where Markov's inequality for positive random variables has been used. The result holds taking the infimum with respect to $t > 0$ on both sides.

4.2 | Closest Separable Systems

For any given structure ϕ, we build the *closest* separable structures $\overline{\phi}$ and $\underline{\phi}$ such that $\underline{\phi} \leq \phi \leq \overline{\phi}$ in terms of misclassification error:

Misclassification Error If we are given two structures ϕ_1 and ϕ_2, the misclassification error is:

$$d(\phi_1, \phi_2) = \sum_{x \in \{0,1\}^N} |\phi_1(x) - \phi_2(x)|. \tag{5}$$

Clearly, d is a metric in the space of all structures. For a rough approximation of ϕ within the set of separable systems we also consider the closest separable ϕ^*:

Closest separable system Given an arbitrary structure ϕ, a closest separable structure ϕ^* is the one that minimizes the misclassification error.

Proposition 10 *The upper bound for ϕ is $\overline{\phi} = 1 - \phi'$, being ϕ' the lower bound of $1 - \phi$.*

Proof By duality, $1 - \overline{\phi} \leq 1 - \phi \leq 1 - \underline{\phi}$. The closest lower bound for $1 - \phi$ is precisely $1 - \overline{\phi} = \phi'$, and the result holds.

Now, we fully characterize ϕ^* and $\underline{\phi}$ using Integer Linear Programming (ILP) formulations. The upper bound $\overline{\phi}$ can be obtained using Proposition 10. The following ILP describes $\underline{\phi}$:

$$\text{min.} \sum_{x \in \{0,1\}^N} d(x) \tag{6}$$

s.t.

$$\underline{\phi}(\vec{0}) = 0, \tag{7}$$

$$\underline{\phi}(\vec{1}) = 1, \tag{8}$$

$$d(x) = \phi(x) - \underline{\phi}(x), \forall x \in \{0,1\}^N \tag{9}$$

$$\sum_{i=1..N} n_i x_i > \underline{\phi}(x), \forall x \in \{0,1\}^N \tag{10}$$

$$\sum_{i=1..N} n_i x_i \leq 1 + M\underline{\phi}(x), \forall x \in \{0,1\}^N \tag{11}$$

$$\underline{\phi}(x) \in \{0,1\}, \forall x \in \{0,1\}^N \tag{12}$$

$$d(x) \in \{0,1\}, \forall x \in \{0,1\}^N \tag{13}$$

$$n_i \geq 0, \forall i \in \{1, \ldots, N\} \tag{14}$$

Where:

- The objective function (6) states that the model will minimize the missclassification error.
- Constraints (7) and (8) establish monotonicity.
- Constraints (9) define the misclassification error $d(x)$ between $\underline{\phi}$ and ϕ.
- Constraints (10) and (11) state that ϕ^* is separable.
- Constraints (12) to (14) define the domains of the decision variables.

Note that the fact that $\underline{\phi}(x) \leq \phi * x) \forall x \in \{0,1\}^N$ arises from the combination of (9) and (13).

Also observe that Constraints (10) are active when $x = (x_1, \ldots, x_N)$ is a pathset of $\underline{\phi}$. In this case (11) is superfluous (choosing the constant M large enough). Constraints (11) define a similar condition for cutsets of ϕ.

The following ILP model provides ϕ^*:

$$\text{min.} \sum_{x \in \{0,1\}^N} d(x) \tag{15}$$

s.t.

$$\phi^*(\vec{0}) = 0, \tag{16}$$

$$\phi^*(\vec{1}) = 1, \tag{17}$$

$$d(x) \geq \phi(x) - \phi^*(x), \forall x \in \{0,1\}^N \tag{18}$$

$$d(x) \geq \phi^*(x) - \phi(x), \forall x \in \{0,1\}^N \tag{19}$$

$$\sum_{j=1..N} n_j x_j > \phi^*(x), \forall x \in \{0,1\}^N \tag{20}$$

$$\sum_{j=1..N} n_j x_j \leq 1 + M\phi^*(x), \forall x \in \{0,1\}^N \tag{21}$$

$$\phi^*(x) \in \{0,1\}, \forall x \in \{0,1\}^N \tag{22}$$

$$d(x) \in \{0,1\}, \forall x \in \{0,1\}^N \tag{23}$$

$$n_i \geq 0, \forall i \in \{1, \ldots, N\} \tag{24}$$

Where:

- $d(x)$ is set to 1 if $\phi(x) \neq \phi^*(x)$.

4.3 | Reliability Bounds

Combining the ILP formulation for $\underline{\phi}$ and Chernoff bounds (Lemma 9), a lower bound for an arbitrary SBS ϕ is produced. Finally, combining the ILP formulation for $\underline{\phi}$, Chernoff bounds and Theorem 12, an upper bound for an arbitrary SBS ϕ is produced. First, we state a technical lemma:

Lemma 11 *The dual of a separable system is also separable.*

Proof If we are given a separable system with hyperplane $H) \sum_{i=1}^N n_i x_i = \alpha_0$, being $\vec{n}$ non-negative, then $H^d) \sum_{i=1}^N n_i x_i = 1 - \alpha_0$. In fact, if we are given a pathset from the dual $x \in \mathcal{P}^d$, we know that $\overline{x} = \vec{1}_N - x$ is a cutset in the original system, and $\sum_{i=1}^N n_i x_i = \sum_{i=1}^N n_i(1 - \overline{x_i}) = 1 - \sum_{i=1}^N n_i \overline{x_i} > 1 - \alpha_0$. A similar calculation holds for pathsets.

Theorem 12 $R_{\mathcal{S}} \geq 1 - R'$, *being* R' *the Chernoff bound for* $(\underline{\phi})^d$.

Proof By duality we know that $\phi^d \leq (\underline{\phi})^d$. Since the dual of a separable system is also separable, we can apply Lemma 9 in order to find an upper bound R' for the reliability of the system $(\underline{\phi})^d$. We get that $R_{\mathcal{S}^d} \leq R'$. Finally, recall that the reliability of a dual is complementary to the reliability of the original system (Lemma 5). Therefore:

$$R_{\mathcal{S}} = 1 - R_{\mathcal{S}^d} \geq 1 - R'. \tag{25}$$

The reader can find an application of these bounds in [11]. As shown in that work, the bounds are not tight for general SBSs, and there is a large room of future work in this research line.

5 | ANALYSIS OF SEPARABLE SYSTEMS

In this section, we first study the hardness of the reliability evaluation for separable systems. Then, we provide two alternative characterizations of these systems.

5.1 | Complexity

Even though separable systems accept an efficient representation, their reliability evaluation is computationally hard:

Theorem 13 *The reliability evaluation of separable systems belongs to the class of $\mathcal{NP}$-Hard problems.*

Proof By reduction from *PARTITION*. Consider an instance of natural numbers $A = \{a_1, \ldots, a_N\}$, and let $s = \sum_{i=1}^{N} a_i$ be the sum over the elements of the list. Let us define $n_i = \frac{a_i}{s}$, $n_{min} = \min_{i=1,\ldots,N}\{n_i\}$, and consider the separable systems $\mathcal{S}_1$ and $\mathcal{S}_2$:

1. The separable system $\mathcal{S}_1$ characterized by the hyperplane $\sum_{i=1}^{N} n_i x_i = \frac{1}{2} + \frac{n_{min}}{2}$;
2. The separable system $\mathcal{S}_2$ characterized by the hyperplane $\sum_{i=1}^{N} n_i x_i = \frac{1}{2}$;

Observe that the difference of the reliability of both systems evaluated at $p = 1/2$ is:

$$
\begin{aligned}
&R_{\mathcal{S}_2}(1/2) - R_{\mathcal{S}_1}(1/2) \\
&= P(\sum_{i=1}^{N} n_i x_i \geq \frac{1}{2}) - P(\sum_{i=1}^{N} n_i x_i \geq \frac{1}{2} + \frac{n_{min}}{2}) \\
&= P(\sum_{i=1}^{N} n_i x_i = \frac{1}{2}) \\
&= \frac{\#\{(x_1, \ldots, x_N) \in \{0,1\}^N : \sum_{i=1}^{N} n_i x_i = \frac{1}{2}\}}{2^N},
\end{aligned}
$$

and the last number is positive if and only if there exists a subset $B \subseteq \{1, \ldots, N\}$ such that $\sum_{i \in B} n_i = \frac{1}{2}$. In that case, if we multiply on both sides by s we get that $\sum_{i \in B} a_i = \frac{s}{2}$, and the answer to *PARTITION* for the list A is YES. Otherwise, the answer to *PARTITION* is NO. Therefore, the reliability evaluation of separable systems is at least as hard as *PARTITION*, and it belongs to the class of $\mathcal{NP}$-Hard problems.

5.2 | Characterizations

A natural question is to characterize separable systems in terms of pathsets and cutsets. Let us denote $CH(\mathcal{P})$ and $CH(C)$ the convex hull of the pathsets and cutsets respectively.

Theorem 14 *An SBS is separable iff $CH(\mathcal{P}) \cap CH(C) = \emptyset$.*

Proof If the intersection is empty, Hahn-Banach separation theorem for convex sets asserts that there exists a hyperplane H that separates those convex sets [20]. As a consequence, $\phi = \phi_H$ for some hyperplane H.

For the converse, we know that the SBS is separable. Therefore, there exists some hyperplane $H)\ \sum_{i=1}^{N} n_i x_i = \alpha_0$ such that $\sum_{i=1}^{N} n_i x_i \leq \alpha_0$ for cutsets, and $\sum_{i=1}^{N} n_i x_i > \alpha_0$ for pathsets. Suppose for a moment that $CH(\mathcal{P}) \cap CH(C) \neq \emptyset$. There exists some element $z \in \mathbb{R}^N$ such that:

$$z = \sum_{j=1}^{l} \alpha_j x^j = \sum_{k=1}^{s} \beta_k y^k, \tag{26}$$

for some states $x^1, \ldots, x^l \in \mathcal{P}$, $y^1, \ldots, y^s \in C$, and non-negative numbers such that $\sum_{j=1}^{l} \alpha_j = \sum_{k=1}^{s} \beta_k = 1$. If we denote $x^j = (x_1^j, \ldots, x_N^j)$ we know that $\sum_{i=1}^{N} n_i x_i^j > \alpha_0$. Therefore, for $z = (z_1, \ldots, z_N)$ we get that:

$$\begin{aligned}\sum_{i=1}^{N} n_i z_i &= \sum_{i=1}^{N} n_i (\sum_{j=1}^{l} \alpha_j x_i^j) \\ &= \sum_{j=1}^{l} \alpha_j [\sum_{i=1}^{N} n_i x_i^j] \\ &> (\sum_{j=1}^{l} \alpha_j)\alpha_0 = \alpha_0.\end{aligned}$$

Analogously, using the fact that $z = \sum_{k=1}^{s} \beta_k y^k$ we get that $\sum_{i=1}^{N} n_i z_i \leq \alpha_0$, which is a contradiction. Therefore we must have $CH(\mathcal{P}) \cap CH(C) = \emptyset$, and the result holds.

By Proposition 14 we have a full geometrical characterization of separable systems, which accept an efficient representation.

In the following, we consider an alternative characterization, in terms of weighted cutsets and pathsets. Consider an arbitrary assignment $n_1, \ldots, n_N$ of non-negative numbers to the respective components of the system. The condition $\sum_{i=1}^{N} n_i x_i \geq \alpha_0$ for all the pathsets is equivalent to finding the pathset x with minimum-cost, $c(x) = \sum_{i:x_i=1} n_i$, and testing if $c(x) \geq \alpha_0$. Analogously, the condition $\sum_{i=1}^{N} n_i y_i < \alpha_0$ for all the cutsets is equivalent to testing whether the cutset y with minimum cost, $c(y) = \sum_{i:y_i=0} n_i$, satisfies the test $S - c(y) < \alpha_0$, where $S = \sum_{i=1}^{N} n_i$ is the cost of the global system. Observe that, for convenience, the cost of a cutset is defined as the sum of the components under failure. In particular, we get the following characterization of separable systems:

Theorem 15 *An SBS is separable if and only if there exists an assignment of non-negative costs to the components $\{n_i\}_{i=1,\ldots,N}$ such that $S < c(y) + c(x)$, being $c(x)$ and $c(y)$ the pathset/cutset with minimum cost respectively.*

Proof First, let us assume that we have a separable SBS with hyperplane $\sum_{i=1}^{N} n_i x_i = \alpha_0$. Using the previous reasoning, the assignment $\{n_i\}_{i=1,\ldots,N}$ verifies $c(x) \geq \alpha_0$ and $S - c(y) < \alpha_0$. Therefore, $S < c(y) + c(x)$.

For the converse, let us fix $\alpha_0 = c(x)$, the pathset with minimum cost. Clearly, the specific pathset x meets the condition $\sum_{i=1}^{N} n_i x_i \geq \alpha_0$; in fact the equality is met. By its definition, the inequality holds for the other pathsets. Finally, we use the fact that $S < c(y) + c(x)$ to verify that the cutset with minimum-cost, y, meets the inequality $\sum_{i=1}^{N} n_i y_i < \alpha_0$. The inequality for the other cutsets is straight since y is a cutset with minimum-cost. Therefore, the SBS is separable, concluding the proof.

6 | SEPARABILITY IN GRAPHS

Our characterization of separable systems has a straight reading in the celebrated all-terminal reliability model.

Separable Graph A graph $G = (V, E)$ is *separable* if there exists an assignment of non-negative real numbers $n_1, \ldots, n_m$ to its m links, and there exists a threshold α such that $c(E') \geq \alpha$ if and only if the spanning subgraph $G' = (V, E')$ is connected.

Let G be a connected graph. Recall that Kruskal algorithm provides efficiently the cost of the minimum spanning tree, $MST(G)$. Furthermore, the cutset with minimum-cost, $m(G)$, is obtained using Ford-Fulkerson algorithm. Therefore, the following corollary of Theorem 15 holds for graphs:

Corollary 16 *A graph is separable iff there exists a feasible assignment $\{n_i\}_{i=1,\ldots,N}$ to the links such that $S < MST(G) + m(G)$, being $MST(G)$ the cost of the minimum spanning tree, $m(G)$ the mincut with minimum capacity, and $S = \sum_{i=1}^{N} n_i$ the sum of the link weights.*

For example, trees and elementary cycles are separable graphs. Indeed, if T_n is a tree with n nodes, a feasible assignment is an identical unit-cost to all the links, since in that case $MST(T_n) = n - 1$, $m(G) = 1$ and the global sum is $S = n - 1 < MST(T_n) + m(T_n)$. Analogously, if C_n denotes the elementary cycle with n nodes, then $S = n < (n - 1) + 2 = MST(C_n) + m(C_n)$, and the same unit-cost assignment works.

Intuitively, if the graph is dense enough, it is not expected to exceed the global cost S of the graphs using the minimum spanning tree and mincut. Our first result deals with the extremal case of complete graphs:

Proposition 17 *Complete graphs $(K_n)_{n\geq 4}$ are nonseparable.*

Proof Consider an arbitrary assignment $\{n_i\}_{i=1,\ldots,n(n-1)/2}$ to the links of K_n, and an arbitrary star-graph $K_{1,n}$ contained in K_n. Since $K_{1,n}$ is connected, its cost is greater than, or equal to the minimum spanning tree, so, $c(K_{1,n}) \geq MST(K_n)$. Furthermore, the complementary links of $K_{1,n}$, or the complementary graph $K_{1,n}^C$, is a cutset (it isolates a single node), so the cost must exceed the mincut: $c(K_{1,n}^C) \geq m(K_n)$. But then, the global cost is $c(K_n) = c(K_{1,n}) + c(K_{1,n}^C) \geq MST(K_n) + m(K_n)$. The conclusion is that $S = c(K_n) \geq MST(K_n) + m(K_n)$ for any feasible assignment, and K_n is nonseparable.

With the following lemmas, we will present a hereditary property of separable graphs, stated in Theorem 20. Consider a simple connected graph $G = (V, E)$. We will consider two different link additions:

- We denote $G_{in} = G + e_{in}$ to the resulting graph after the addition of an internal link $e_{in} = \{u_1, u_2\}$, where $u_1, u_2 \in V$.
- We denote $G_{out} = G + e_{out}$ to the resulting graph after the addition of an external link $e_{out} = \{u_1, u_2\}$, where $u_1 \in V$ but $u_2 \notin V$.

Observe that $G + e_{in}$ and G share an identical node-set V, while the node-set for $G + e_{out}$ is $V \cup \{u_2\}$.

Lemma 18 *If G is nonseparable then G_{out} is nonseparable.*

Proof Suppose for a moment that there exists a feasible assignment $\{n_i\}_{i=1,\ldots,N+1}$ for G_{out}. Then:

$$\begin{aligned}(\sum_{i=1}^{N} n_i) + n_{N+1} &< MST(G_{out}) + m(G_{out}) \\ &= MST(G) + n_{N+1} + \min\{m(G), n_{N+1}\} \\ &\leq MST(G) + n_{N+1} + m(G),\end{aligned}$$

and $\{n_i\}_{i=1,\ldots,N}$ would be a feasible assignment for G, which is a contradiction. Therefore, G_{out} is nonseparable.

Lemma 19 *If G is nonseparable then G_{in} is nonseparable.*

Proof Suppose for a moment that there exists a feasible assignment $\{n_i\}_{i=1,\dots,N+1}$ for G_{in}. Then:

$$\begin{aligned}(\sum_{i=1}^{N} n_i) + n_{N+1} &< MST(G_{in}) + m(G_{in}) \\ &\leq MST(G) + m(G) + n_{N+1},\end{aligned}$$

and $\{n_i\}_{i=1,\dots,N}$ would be a feasible assignment for G, which is a contradiction. Therefore, G_{in} is nonseparable.

Observe that Lemma 19 informally states that graphs with more density are nonseparable. Using the counter-reciprocal of Lemmas 18 and 19, we obtain the following:

Theorem 20 *Separability is a hereditary property in graphs.*

Proof Reading the counter-reciprocal of Lemma 19, we know that the deletion of one or several links from a separable graph is also separable. By Lemma 18, we also know that a node-deletion in a separable graph (with the intermediate deletion of links using Lemma 19) is also separable. Combining node and link deletions, an arbitrary subgraph is obtained, and it must be separable as well.

Lemma 21 *If G is separable, G_{out} is also separable.*

Proof Consider a feasible assignment $\{n_i\}_{i=1,\dots,N}$ for G, where $S < MST(G) + m(G)$ holds. Let us consider an extended assignment with n_{N+1} for the external link, such that $n_{N+1} > m(G)$. Then:

$$\begin{aligned}S + n_{m+1} &< (MST(G) + n_{N+1}) + m(G) \\ &= MST(G_{out}) + \min\{m(G), n_{N+1}\} \\ &= MST(G_{out}) + m(G_{out}),\end{aligned}$$

and $\{n_i\}_{i=1,\dots,N+1}$ is a feasible assignment for G_{out}.

Corollary 22 *Cycles with arborescences are separable graphs*

Proof We know that elementary cycles are separable. The result follows by the addition of one or several trees hanging to different nodes from the first cycle. Supported by Lemma 21, the separability is preserved by the addition of those links.

Figure 1 depicts Monma graphs. They have two degree-3 nodes connected by 3 node-disjoint paths. We already know that an arbitrary subgraph of Monma is separable, since it is a graph with arborescence (or disconnected graphs, which are trivially separable graphs). However, we will see that Monma graphs are minimally nonseparable graphs. Clyde Monma et. al. used these graphs to design minimum cost biconnected metric networks [22]. Furthermore, they attain the maximum reliability among all the graphs with p nodes and $q = p + 1$ links [23].

Lemma 23 *Monma graphs are nonseparable*

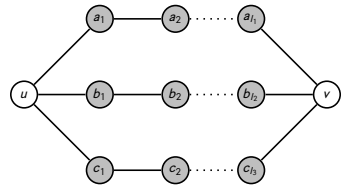


FIGURE 1 Monma graph M_{l_1+1,l_2+1,l_3+1}.

Proof Consider an arbitrary order for the links of Monma graph, and the rule $\phi(x) = 1$ iff the Monma subgraph given by the links in x is connected. We will find a convex combination of pathsets and cutsets with identical result. Consider the four links $e_1 = \{u, a_1\}$, $e_2 = \{a_1, a_2\}$, $e_3 = \{u, b_1\}$ and $e_4 = \{b_1, b_2\}$ from Figure 1. Let us denote $1_{e_i,e_j}$ the binary word that is set to 1 in all the bits but 0 in the positions corresponding to the links e_i and e_j. Consider the following identity:

$$\frac{1}{2}(1_{e_1,e_2} + 1_{e_3,e_4}) = \frac{1}{4}(1_{e_1,e_3} + 1_{e_1,e_4} + 1_{e_2,e_3} + 1_{e_2,e_4}) \tag{27}$$

On one hand, we have a convex combination of cutsets. On the other, a convex combination of pathsets. By Theorem 14, Monma graphs are nonseparable.

Recall that a node v in a graph G is a cut-point if $G - v$ has more components than G. A connected graph is biconnected if it has no cut-points. The addition of an ear in a graph G is the addition of an external elementary path between two different nodes from G. Frederickson-Jàjà characterization theorem asserts that there exists an ear decomposition of all biconnected graphs, such that $G = C_s \cup H_1 \cup H_2 \cup \cdots \cup H_r$, C_s is an elementary cycle and H_i is the addition of an ear to the previous graph [24]. This structural characterization of biconnected graphs lead us immediately to the following:

Theorem 24 *All biconnected graphs that are not elementary cycles are nonseparable.*

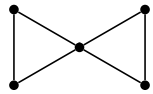
Proof As the base-step, we know by Lemma 23 that Monma graphs are nonseparable. If G is biconnected and it is not an elementary cycle, then it has the addition of at least one ear of a cycle. Therefore, it has Monma as a subgraph. Therefore, Theorem 20 asserts that G cannot be separable.

Recall that the link-connectivity of a graph G is the least number of links that must be removed in order to disconnect G. The Bowtie-graph consists of two triangles meeting in a common point (see Figure 2). This is the smallest graph with link connectivity 2 that is not biconnected, since the kissing-point is a cut-point. As a consequence, it is natural to decide the separability of this graph:

Lemma 25 *The Bowtie-graph B is nonseparable*

Proof Consider an arbitrary assignment $\{n_i\}_{i=1,\ldots,6}$ for the links. We consider an assignment $n_1 \leq n_2 \leq n_3$ in the left triangle, and $n_4 \leq n_5 \leq n_6$ in the right triangle. Therefore $MST(B) = S - n_3 - n_6$, and $m(B) = \min\{n_1 + n_2, n_4 + n_5\} \leq \min\{2n_2, 2n_5\} \leq n_3 + n_6$. This implies that $MST(B) + m(B) \leq S$ for all possible assignments in B, and B has no feasible assignment.

An analogous reasoning leads to the following generalization:



Bowtie

FIGURE 2 Bowtie-graph B.

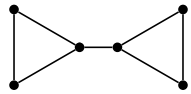


FIGURE 3 Glasses-graph B_e.

Corollary 26 *Two kissing cycles are nonseparable.*

A further generalization recalls Frederickson-Jàjà characterization: G is a bridgeless graph if and only if $G = C_s \cup H_1 \cup H_2 \cup \cdots \cup H_r$, C_s is an elementary cycle and H_i is the addition of an ear or a kissing cycle to the previous graph. The following result is analogous to Theorem 24:

Corollary 27 *Bridgeless graphs are nonseparable, except for elementary cycles.*

In order to fully characterize separable graphs, we need to study graphs that have at least one bridge $e \in G$. We already know that all the links in a tree are bridges, and they are separable graphs. Furthermore, cycles with arborescences are separable as well. Let us proceed our analysis with two triangles linked by a single bridge e, a graph called as Glasses-graph B_e.

Lemma 28 *The Glasses-graph B_e is nonseparable.*

Proof The reasoning is identical to the Bowtie-graph. Consider an assignment $\{n_i\}_{i=1,\ldots,7}$ as in the Bowtie-graph, but n_7 is the assignment for the bridge e. Therefore:

$$\begin{aligned} MST(B_e) + m(B_e) &= (S - n_3 - n_6) \\ &\quad + \min\{n_1 + n_2, n_4 + n_5, n_7\} \\ &\leq S, \end{aligned}$$

since $\min\{n_1 + n_2, n_4 + n_5, n_7\} \leq n_3 + n_6$, and the last inequality was already proved for the Bowtie-graph.

A slight generalization is possible:

Corollary 29 *Two cycles linked by an elementary path are nonseparable.*

We are in conditions to fully characterize separable graphs:

Theorem 30 *A graph G is separable iff G falls into one of the four categories:*

1. *G is not connected;*
2. *G is a tree;*
3. *G is an elementary cycle; or*
4. *G is an elementary cycle with arborescences.*

Proof In order to prove the converse, we test case by case that the graph is separable:

1. If G is disconnected, all of its configurations are cutsets and the reliability is null. In this case, the inequality $\sum_{i=1}^{N} x_i > 2N$ is not satisfied by any binary vector $x = (x_1, \ldots, x_N)$, and the graph is separable.
2. If G is a tree T_N with N links, the evidence is the hyperplane $\sum_{i=1}^{N} x_i \geq N$ (or a unit-assignment is feasible).
3. If $G = C_N$ is an elementary cycle, the evidence is the inequality $\sum_{i=1}^{N} x_i \geq N - 1$.
4. If G is a tree with arborescences, Lemma 22 states that G is separable.

Finally, let G be a separable graph, and assume G is connected. We know by Corollary 27 that G must have a bridge. Combining Theorem 20 and Corollary 24, G cannot have any bridgeless subgraph different than elementary cycles. Combining Corollaries 26 and 29, G cannot have two cycles (either they are kissing or connected by a path). Therefore, G is either a tree, an elementary cycle of an elementary cycle with arborescences.

Corollary 31 *The all-terminal reliability evaluation of separable graphs belong to the class $\mathcal{P}$ of polynomial-time problems.*

Proof The analysis is straight. Let G be a separable graph:

1. If G is not connected $R(G) = 0$.
2. If $G = T_N$ a tree with N links with independent reliabilities $(p_e)_{e \in T_N}$ then $R(G) = \prod_{e \in T_N} p_e$.
3. If $G = C_N$,

$$R(C_N) = \prod_{e \in C_N} p_e + \sum_{e \in C_N} (1 - p_e) \prod_{e' \neq e} p_{e'}.$$

4. Finally, if G is an elementary cycle with arborescences: $G = C_l \cup T_s$, being T_s union of trees pending from the cycle C_l. Therefore, $R(G) = R(C_l) \times \prod_{e \in T_s} p_e$.

The reader can appreciate that the reliability computation is a product, or a sum of products of the elementary link reliabilities. Therefore, the number of operations involved are linear, or quadratic, in the number of links.

The corank of a graph is the number of independent cycles. In a connected graph with n nodes and m links, its corank is precisely $c(G) = m - n + 1$. It is worth to remark that Theorem 30 can be re-stated in terms of corank: a connected graph G is separable if and only if its corank is either 0 or 1.

We close this section discussing an interplay between a combinatorial optimization called the Network Utility Problem (NUP) and separable graphs. First, observe that an arbitrary spanning tree of a connected graph G has $n-1$ links. Therefore, the corank of a graph is precisely the number of *additional links that we must pay* to build the graph G, starting from a minimally-connected graph. In terms of communication, the corank of G represents *redundancy*. At the cost of redundancy, the resulting network can be robust under a certain amount of link failures. The profit is the link connectivity $\lambda(G)$, which represent the lowest number of links that should be removed in order to disconnect G. As a consequence,

the *utility* of a graph, $u(G)$, is the difference between the connectivity and the corank: $u(G) = \lambda(G) - c(G) = \lambda - m + n - 1$. In [25], the authors formally proved the following

Theorem 32 *The graphs with maximum utility are trees and cycles. Their utility value is* 1. *There is no other graph with maximum utility.*

Corollary 33 *All the graphs with maximum utility are separable graphs.*

The all-terminal reliability polynomial under identical elementary reliabilities in the links r is

$$R_G(r) = \sum_{i=\lambda(G)}^{c(G)-1} n_i(G) p^{m-i} (1-p)^i + \tau(G) p^{n-1} (1-p)^{m-n+1}, \tag{28}$$

being $n_i(G)$ the number of connected subgraphs of G with precisely $m - i$ links, and $\tau(G)$ the tree-number of G, which is known using Matrix-Tree Kirchhoff theorem [26]. Therefore, the number of unknowns is precisely the number of terms involved in the summation: $c(G) - \lambda$. The only cases where there are no terms in the sum occur either when $c(G) - \lambda = -1$, exactly in trees and cycles, or when $c(G) - \lambda = 0$, only in an elementary cycle with arborescence, K_4, Kite-graph and Bowtie-graph [25]. These graphs are considered as the simplest in terms of reliability analysis. Indeed, in [25] the authors define the *level of difficulty* of a graph as the difference $d(G) = c(G) - \lambda - 1$, and a graph is easy if and only if $d(G) \leq 0$:

Corollary 34 *All separable graphs are easy graphs.*

The reader can observe that the graphs with maximum utility $u(G)$ are the easiest graphs, with the minimum level of difficulty $d(G)$.

7 | DISCUSSION AND CONCLUDING REMARKS

7.1 | Discussion

A natural extension of our prior analysis is a classification of nonseparable systems.

Let $\mathcal{S} = (S, r, \phi)$ be an arbitrary SMBS, and consider its corresponding 0-1 labels of the vertices of a hypercube Q_N in the Euclidean space $\mathbb{R}^N$.

Level of Separability The *level of separability of* $\mathcal{S}$ is the least positive integer d such that there exists positive separator hyperplanes $\pi_1, \ldots, \pi_d$, where all the pathsets of $\mathcal{S}$ lie on the same side as the unit vector for all the hyperplanes, and all the cutsets do not meet the previous condition, at least for one hyperplane.

Proposition 35 *Let $\mathcal{S}$ be an arbitrary SMBS, and let $mc = |\mathcal{MC}|$ be the number of all its mincuts. Therefore, the level of separability d verifies $d \leq mc$.*

Proof Assume that $x^1, \ldots, x^{mc}$ is the list of all the mincuts of $\mathcal{S}$. Consider the sets $S_i = \{j : x^i_j = 0\}$, that represent the non-operational states for the mincut x^i. Observe that the mincut x^i does not meet the inequality $\pi_i : \sum_{j \in S_i} x_j \geq 1$. Furthermore, the hyperplanes $\pi_1, \ldots, \pi_{mc}$ meet the definition 7.1, and the result follows.

Proposition 35 shows that the level of separability will always be well defined for any arbitrary SMBS, thus it is an alternative way to classify a notion of *difficulty* in the reliability evaluation for SMBSs.

If we return to the all-terminal reliability model, we know from Theorem 30 all the graphs with level of separability $d = 1$ (i.e., all separable graphs). We can observe that the Bowtie-graph, the Glasses-graph and Monma represent minimally nonseparable cases. For a better understanding of definition 7.1, we find the level of separability in these minimally nonseparable cases in the following paragraphs.

Let us denote x_1, x_2, x_3 and y_1, y_2, y_3 to the states of the links for the Bowtie-graph, corresponding to both triangles (see Figure 2). All pathsets must have at least two links from every triangle, and the following 2 hyperplanes determine pathsets:

$$x_1 + x_2 + x_3 \geq 2$$
$$y_1 + y_2 + y_3 \geq 2.$$

Since we know that the Bowtie-graph is nonseparable, $d > 1$, and since the previous hyperplanes fulfill the definition, the level of separability for the Bowtie-graph is $d = 2$.

Analogously, if we link both triangles with a new link z, we get the Glasses-graph. A slight modification of the hyperplanes serve as an evidence that the Glasses-graph has level of separability $d = 2$:

$$x_1 + x_2 + x_3 + 3z \geq 5$$
$$y_1 + y_2 + y_3 \geq 2.$$

Observe that we *force* the link z to be operational, adding the term $3z$ in the first hyperplane. Finally, let us consider Monma graph $M_{2,2,1}$ from Figure 1, where the three paths have respective lengths 2, 2 and 1, and the respective links from each path are sequentially identified with the binary states x_1, x_2, y_1, y_2 and z. The reader is invited to check that the level of separability in Monma graph $M_{2,2,1}$ is also $d = 2$, and the following pair of hyperplanes works:

$$10x_1 + 10x_2 + y_1 + y_2 + z \geq 12$$
$$x_1 + x_2 + 10y_1 + 10y_2 + z \geq 12.$$

Currently, there is no constructive algorithm to produce the minimum number of hyperplanes for an SMBS. We wish to develop a complementary theory to the one presented in Section 6 for separable graphs, but finding the correct level of separability for any given graph. Inspired by Theorem 31, we promote the following:

Conjecture 1 *Let d be a fixed positive integer. Then, the all-terminal reliability evaluation of graphs with level of separability d belongs to the class $\mathcal{P}$ of polynomial-time problems.*

7.2 | Concluding Remarks

In this work, we study the reliability evaluation of stochastic binary systems (SBS), and its impact in the celebrated all-terminal reliability model.

An efficient representation of separable systems is here presented, and a full characterization of these special systems. The major strength of separable systems is their efficient representation. The major shortcoming is that the reliability evaluation is as hard as an arbitrary SBS. Supported by duality, separability and large deviation theory,

reliability bounds were presented for arbitrary SBSs. The bounds are not tight in general, and this fact motivates further research.

Separable systems accept polynomial-time reliability evaluation when restricted to the all-terminal reliability model. This result was discovered using functional analysis and feasible functionals from the links of a graph, meeting separability constraints. This interplay between SBSs and Functional Analysis should be further studied.

As future work, we would like to establish Conjecture 1 for a better understanding of nonseparable systems, and the interplay between general SBSs and the all-terminal reliability model, which has a wide spectrum of applications.

ACKNOWLEDGMENT

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos* and by Project MATHAMSUD 19-MATH-0 *RareDep*.

REFERENCES

[1] Lee HW, Lee K, Modiano E. Maximizing Reliability in WDM Networks Through Lightpath Routing. IEEE/ACM Transactions on Networking 2014 Aug;22(4):1052–1066.

[2] Barrera J, Cancela H, Moreno E. Topological optimization of reliable networks under dependent failures. Operations Research Letters 2015;43(2):132 – 136.

[3] Barta J, Montemanni. Wireless multicasting under probabilistic node failures: a heuristic approach. Optimization and Engineering 2012;13(4):705–726.

[4] Markowetz F, Boutros M. Linking Phenotypes and Genotypes. Cambridge Series in Systems Genetics, Cambridge University Press; 2015.

[5] Song X, Zhai Z, Liu Y, Han J. A stochastic approach for the reliability evaluation of multi-state systems with dependent components. Reliability Engineering & System Safety 2018;170(Supplement C):257 – 266.

[6] Lin M. An Efficient Algorithm for Computing the Reliability of Stochastic Binary Systems. IEICE Transactions 2004;87-D(3):745–750.

[7] Canale E, Cancela H, Piccini J, Robledo F, Romero P, Rubino G, et al. Recursive Variance Reduction method in stochastic monotone binary systems. In: 2015 7th International Workshop on Reliable Networks Design and Modeling (RNDM); 2015. p. 135–141.

[8] Canale E, Cancela H, Robledo F, Romero P, Sartor P. Diameter constrained reliability: Complexity, distinguished topologies and asymptotic behavior. Networks 2015;66(4):296–305.

[9] Levitin G. Computational Intelligence in Reliability Engineering: Evolutionary Techniques in Reliability Analysis and Optimization. Studies in Computational Intelligence, Springer Berlin Heidelberg; 2006.

[10] Cancela H, Guerberoff G, Robledo F, Romero PG. Reliability Maximization in Stochastic Binary Systems. In: 14th International Conference on the Design of Reliable Communication Networks (DRCN 2018) Paris, France; 2018. .

[11] Cancela H, Ferreira G, Guerberoff G, Robledo F, Romero P. Building Reliability Bounds in Stochastic Binary Systems. In: 2018 10th International Workshop on Resilient Networks Design and Modeling (RNDM); 2018. p. 1–7.

[12] Ball MO. Computational Complexity of Network Reliability Analysis: An Overview. IEEE Transactions on Reliability 1986 aug;35(3):230 –239.

[13] Romero P. Duality in stochastic binary systems. In: 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM); 2016. p. 85–91.

[14] Ball MO. Complexity of network reliability computations. Networks 1980;10(2):153–165.

[15] Petingi L, Rodriguez J. Reliability of networks with delay constraints. In: Congressus Numerantium, vol. 152; 2001. p. 117–123.

[16] Canale E, Cancela H, Robledo F, Romero P, Sartor P. Full complexity analysis of the diameter-constrained reliability. International Transactions in Operational Research 2015;22(5):811–821.

[17] Cancela H, Urquhart ME. RVR simulation techniques for residual connectedness network reliability evaluation. IEEE Transactions on Computers 2002;51(4):439–443.

[18] Cook SA. The complexity of theorem-proving procedures. In: Proceedings of the third annual ACM symposium on Theory of computing STOC '71, New York, NY, USA: ACM; 1971. p. 151–158.

[19] Rosenthal A. Computing the Reliability of Complex Networks. SIAM Journal on Applied Mathematics 1977;32(2):384–393.

[20] Conway JB. A Course in Functional Analysis. Graduate Texts in Mathematics, Springer New York; 1994.

[21] Cramér H. Half a century with probability theory: some personal recollections. Annals of Probability 1976;4:509–546.

[22] Monma C, Munson B, Pulleyblank W. Minimum-weight two-connected spanning networks. Mathematical Programming 1990;46(1-3):153–171.

[23] Boesch FT, Li X, Suffel C. On the existence of uniformly optimally reliable networks. Networks 1991;21(2):181–194.

[24] Fredrickson GN, JaJa JF. On the relationship between the biconnectivity augmentation and traveling salesman problem. Theoretical Computer Science 1982;19:189 – 201.

[25] Canale E, Romero P, Rubino G, Warnes X. Network utility problem and easy reliability polynomials. In: 2016 8th International Workshop on Resilient Networks Design and Modeling (RNDM); 2016. p. 79–84.

[26] Biggs N. Algebraic Graph Theory. Cambridge Mathematical Library, Cambridge University Press; 1993.

Chapter 8

Model Construction in Stochastic Binary Systems

In this chapter we present a second stage of the work introduced in Chapter 7. Here a model construction methodology is proposed, supported by a random sample of the space-state that considers Support Vector Machine (SVM) and a binary classification.

Model Construction in Stochastic Binary Systems

Mathias Bourel, Héctor Cancela, Gustavo Guerberoff, Franco Robledo, Pablo Romero and Luis Stábile
Facultad de Ingeniería
Universidad de la República
Julio Herrera y Reissig 565, PC 11300
Montevideo, Uruguay
Emails: (mbourel,cancela,gguerber,frobledo,promero,lstabile)@fing.edu.uy

***Abstract*—A Stochastic Binary System (SBS) is a mathematical model of a multi-component on-off system, where its components are subject to random failures. The reliability evaluation of an SBS is a common target in the research community, since it includes the traditional network reliability measures, which have many applications.**

Several works from the scientific literature assume that the rule that determines the system operation is fully known, and that the failures are independent with known probabilities. However, the number of feasible states for a system with N on-off components is 2^N, and an exhaustive list of all the states is computationally prohibitive. Furthermore, the reliability evaluation of an arbitrary SBS belongs to the class of $\mathcal{NP}$-Hard problems.

In this work, we outline the main challenges in the reliability evaluation of an SBS, and recent progress in the field. Additionally, we propose a model construction of an SBS supported by a random sample of the space-state that considers Support Vector Machine (SVM) and binary classification. We believe that this approach is not only more realistic, but it is also a point of departure to discuss the traditional independence assumption in general SBS.

***Index Terms*—Stochastic Binary Systems, Network Reliability, Support Vector Machine, Machine Learning.**

I. Motivation

The term reliability stands for the probability of correct operation of a system. Given the importance of networks in our lives, network reliability analysis is a mature field of knowledge, and the literature is vast. The common approach is to consider random graphs with either node or link failures, depending on the specific application. Stochastic Binary Systems (SBS) represent a more abstract setting, where the components subject to random failures are not necessarily configured or related by a network. Real examples are provided by vehicles, redundant systems, robots or systems of systems. Pioneer works by Provan, Colbourn [1] and Rosenthal [2] confirm the hardness of the network reliability evaluation. The reader is invited to consult the monograph [3] for an authoritative work in the field of network reliability. The diameter-constrained reliability is also hard, even for all the diameters $d \geq 3$, see [4]. As corollary, the reliability evaluation of general SBS is also hard, since it subsumes the hardness of the previous sub-models. What is more, the rule that determines the operation or failure of the system for a given configuration has an exponential domain. Therefore, the challenges to extend a theory on stochastic binary systems are noteworthy.

This work is part of an international Research & Development project entitled *Dynamic Stochastic Binary Systems*. From a high level point of view, this project includes the following three stages:

1) Understand the combinatorics and reliability of Stochastic Binary Systems.
2) Define realistic models with dependent failures.
3) Introduce time and dynamism in the previous models.

A concrete example of a dynamic SBS with dependent failures on its components is a car, where a failure on a wheel modifies the failure probability of the other wheels, and the concept of reliability is understood as a correct operation of a trajectory, or a target distance. So far, our achievements can be summarized in the following items:

1) A better understanding of the combinatorics and structure of SBS.
2) We introduced special sub-systems with an efficient representation.
3) We found reliability bounds for general SBS, under the independence assumption.

Recent works confirm that there is a special class of SBS, called *separable systems*, that accept an efficient representation, and the truth-table can be found by an inner product in the Euclidean space [5]. Furthermore, this class of systems can be fully characterized using functional analysis, and we have built reliability bounds for general SBS using the duality of separable systems. The present work represent a first approach to address the second stage of the above mentioned research project. The contributions of this paper can be summarized in the following items:

1) A model construction methodology is proposed for the structure of stochastic binary systems.
2) A full reliability analysis and model construction is offered for the distinguished systems.
3) The hardness of the reliability evaluation of separable systems is established.
4) The concepts of separability and reliability are discussed for the models under study.

This paper is organized as follows. Section II formally presents the problem under study. At the end of this section we establish the hardness of the reliability evaluation of

separable systems. Distinguished models are studied in Section III inspired by real-life applications (redundant systems, recovery/failure systems). They will be the focus of this work. This section also includes the reliability analysis and a discussion of separability for each model. The selected machine learning techniques for the model construction is outlined in Section IV. Section V presents the misclassification error, and Section VI contains concluding remarks and trends for future work.

II. Concepts

The following terminology is adapted from [6].

Definition 1 (Stochastic Binary System)**.** A *stochastic binary system* is a triad (S, r, ϕ):

- $S = \{1, \ldots, N\}$ is a ground set of *components*,
- $r = (r_1, \ldots, r_N)$ are their *elementary reliabilities*, and
- $\phi : \{0,1\}^N \to \{0,1\}$ is the *structure*.

The concept of reliability is generalized to arbitrary stochastic binary systems.

Definition 2 (Reliability/Unreliability)**.** Let $\mathcal{S} = (S, p, \phi)$ be a stochastic binary system, and consider a random vector $X = (X_1, \ldots, X_N)$ with independent coordinates governed by Bernoulli random variables such that $P(X_i = 1) = r_i$. The *reliability* of $\mathcal{S}$ is the probability of correct operation of the system:

$$R_{\mathcal{S}} = P(\phi(X) = 1) = E(\phi(X)) = \sum_{x:\phi(x)=1} P(X = x). \quad (1)$$

The *unreliability* of $\mathcal{S}$ is $U_{\mathcal{S}} = 1 - R_{\mathcal{S}}$.

A stochastic binary system is *homogeneous* if the elementary reliabilities are identical (i.e., $r_i = r$ for all i).

Definition 3 (Pathsets/Cutsets)**.** Let $\mathcal{S} = (S, r, \phi)$ be a stochastic binary system. A possible state or configuration $x \in \{0,1\}^N$ is a *pathset* (resp. *cutset*) if $\phi(x) = 1$ (resp., if $\phi(x) = 0$).

Definition 4 (Stochastic Monotone Binary System (SMBS))**.** The triad $\mathcal{S} = (S, r, \phi)$ is a *stochastic monotone binary system* if the structure function $\phi : \{0,1\}^N \to \{0,1\}$ is monotonically increasing, $\phi(\overline{0}_N) = 0$ and $\phi(\overline{1}_N) = 1$.

Observe that SMBS represent *well-behaved* SBS, in the sense that, given a working configuration, the system can fail after the removal of some components, but can not fail if some failed components start to work. Additionally, the system does not work if it has no operational components, and the full system works.

Definition 5 (Minpaths/Mincuts/Rays)**.** Let $\mathcal{S} = (S, r, \phi)$ be an SMBS:

- A pathset x is a *minpath* if $\phi(y) = 0$ for all $y < x$.
- A cutset y is a *mincut* if $\phi(x) = 1$ for all $x > y$.
- The x-ray is the set $S_x = \{y \in \{0,1\}^N : y \geq x\}$.

An SMBS is fully characterized by its mincuts (or its minpaths). In fact, if we are given the complete list of minpaths, then the complete list of pathsets is precisely the union of the x-rays for every minpath x. Observe that the representation of an SBS requires a truth-table (i.e., the structure function) with 2^N rows. In order to find an efficient representation, the following concept was introduced in [5]:

Definition 6 (Separable System)**.** An SBS is *separable* if the cutsets/pathsets can be separated by some hyperplane.

Since a hyperplane in the Euclidean space can be fully determined by a normal vector $\overrightarrow{n}$ and a point P, the structure evaluation requires only N multiplications for separable systems, and there is no need to store 2^N rows as in the general case. Separable systems can be fully characterized using pathsets and cutsets. Let us denote $CH(\mathcal{P})$ and $CH(\mathcal{C})$ the convex hull of the pathsets and cutsets respectively.

Proposition 1. *An SBS is separable iff* $CH(\mathcal{P}) \cap CH(\mathcal{C}) = \emptyset$.

The proof exploits Hahn-Banach separation theorem for compact/convex sets [7]. See [5] for a proof. The following examples provide an insight of the different applications of stochastic binary systems to network reliability.

1) *K-Terminal Reliability*: the ground-set is precisely the links of a simple graph. The system is up if the terminal-set K belongs to the same connected component [8].
2) *All-Terminal Reliability*: all nodes belong to the terminal-set K.
3) *Source-Terminal Reliability*: choose $K = \{s, t\}$.
4) Diameter Constrained Reliability: a diameter constraint d is added to the K-Terminal Reliability. The system is up if every pair of terminals are connected by paths whose length is not greater than the diameter [9], [4].
5) *Node-Reliability*: the ground set is the set of the nodes of a simple graph. The system is up if the resulting random graph is connected.

There exists an interplay between SBS and propositional logic. Recall that a theorem-proving procedure is the first $\mathcal{NP}$-Complete decision problem established by Stephen Cook [10]. In other words, the recognition of a tautology is a hard decision problem from propositional logic.

Theorem 1. *The reliability evaluation of an arbitrary SMBS belongs to the class of $\mathcal{NP}$-Hard problems.*

Proof. Arnie Rosenthal formally proved that the reliability evaluation for the K-terminal reliability model belongs to the class of $\mathcal{NP}$-Hard computational problems [2]. Since K-Terminal is a particular SMBS, the result follows by inclusion. □

Corollary. *The reliability evaluation of an arbitrary SBS belongs to the class of $\mathcal{NP}$-Hard problems.*

Theorem 2. *The reliability evaluation of separable systems is $\mathcal{NP}$-Hard.*

Proof. By reduction from $PARTITION$. Consider an instance of natural numbers $A = \{a_1, \ldots, a_N\}$, and let

$s = \sum_{i=1}^{N} a_i$ be the sum over the elements of the list. Let us consider the separable systems $\mathcal{S}_1$ and $\mathcal{S}_2$:

1) $\mathcal{S}_1$ characterized by the hyperplane $\sum_{i=1}^{N} n_i x_i = \frac{1}{2} + \frac{1}{2s}$;
2) $\mathcal{S}_2$ characterized by the hyperplane $\sum_{i=1}^{N} n_i x_i = \frac{1}{2}$;

The difference of the reliability of both systems, considering Bernoulli independent random variables with parameter $p = 1/2$, is:

$$\begin{aligned} & R_{\mathcal{S}_2}(1/2) - R_{\mathcal{S}_1}(1/2) \\ &= P\left(\sum_{i=1}^{N} n_i x_i \geq \frac{1}{2}\right) - P\left(\sum_{i=1}^{N} n_i x_i \geq \frac{1}{2} + \frac{1}{2s}\right) \\ &= P\left(\sum_{i=1}^{N} a_i x_i \geq \frac{s}{2}\right) - P\left(\sum_{i=1}^{N} a_i x_i \geq \frac{s}{2} + \frac{1}{2}\right) \\ &= P\left(\sum_{i=1}^{N} a_i x_i = \frac{s}{2}\right) \\ &= \frac{\#\{(x_1, \ldots, x_N) \in \{0,1\}^N : \sum_{i=1}^{N} s_i x_i = \frac{s}{2}\}}{2^N}, \end{aligned}$$

and the last number is positive if and only if there exists a subset $B \subseteq \{1, \ldots, N\}$ such that $\sum_{i \in B} a_i = \frac{s}{2}$. This means the valure will be possitive iff the answer to *PARTITION* for the list A is YES. Otherwise, the answer to *PARTITION* is NO. Therefore, the reliability evaluation of separable systems is at least as hard as *PARTITION*, and it belongs to the class of $\mathcal{NP}$-Hard problems. □

Observe that separable systems accept an efficient representation, but their reliability evaluation is still $\mathcal{NP}$-Hard.

Proposition 2. *A separable system with positive normal vector is an SMBS.*

Proof. If $x \leq y$ and $(x - P_o)\overrightarrow{n} \geq 0$, then $(y - P_o)\overrightarrow{n} \geq 0$. Therefore, $\phi(x) \leq \phi(y)$, and we have an SMBS as desired. We have denoted $\overrightarrow{n}$ the normal vector, whose components are nonnegative, and P_o a point in the plane. □

We can appreciate that separable systems with positive normal vectors are SMBS; but the converse is false. The smallest counterexample (SC) is the SBS with only two minpaths: $m_1 = (0, 1, 0, 1)$ and $m_2 = (1, 0, 1, 0)$. Note that $c_1 = (1, 1, 0, 0)$ and $c_2 = (0, 0, 1, 1)$ are mincuts. Therefore, $(1/2, 1/2, 1/2, 1/2) = (m_1 + m_2)/2 = (c_1 + c_2)/2$ belongs to the convex hull of pathsets and cutsets, so by Proposition 1, SC is not separable.

III. Models

We will study the following SBS as case examples:

($M1$) *k-out-of-N*, in the homogeneous case with individual survivability $p \in [0, 1]$.
($M2$) *Pattern*: $\phi(x) = 1$ iff $w \subset x$, where $w = 11111$.
($M3$) *Random hyperplane*: pick the coordinates of $\overrightarrow{n}$ and P_o uniformly at random in $[0, 1]$.
($M4$) *Two random hyperplanes*: pick two normal vectors, $\overrightarrow{n_1}$ and $\overrightarrow{n_2}$, and two points, P_o and Q_o, as in the previous example. $\phi(x) = 1$ iff $x\overrightarrow{n_1} > P_o\overrightarrow{n_1}$ and $x\overrightarrow{n_2} > Q_o\overrightarrow{n_2}$.
($M5$) *Event in a Chain*: consider a transition 2×2 matrix M, starting distribution $\pi = (1/2, 1/2)$ and two states $\{0, 1\}$. The system works if the path contains 11 (the state 1 is visited consecutively).
($M6$) *SC with noise*: $\phi(x) = 1$ iff the last 4 bits $x' = (x_{N-3}, x_{N-2}, x_{N-1}, x_N)$ meet either $x' \geq m_1$ or $x' \geq m_2$, being $m_1 = (0, 1, 0, 1)$ and $m_2 = (1, 0, 1, 0)$.

Let us first discuss the separability of each model. Model 1 is simple to understand. It is a separable SMBS, with a normal vector with identical coordinates $n_i = 1$. Model 2 is not separable: observe that the *zig-zag* states $x = 1010...$ and its complementary one $\overline{x}$ are cutsets. However, the states $y = w00000...$ and $\overline{y}$ are pathsets, but $(x + \overline{x})/2 = (y + \overline{y})/2$. As a consequence, the convex hulls have at least a common element, and by Proposition 1, Model 2 (Pattern) is not separable. By construction, Model 3 is separable. Model 4 is not separable, since the union of two half-spaces is not necessarily a half-space. Model 5 is not separable (the reasoning is analogous to Pattern, but using $w = 11$). Finally, Model 6 is not separable, since SC is the smallest nonseparable example. Let us proceed with the reliability analysis of each individual model. The reliability of Model 1 can be found by means of a Binomial distribution (a sum of N independent Bernoulli random variables). If u_N denotes the unreliability of Model 2 with N components, then the sequence (u_n) respects the initial condition $u_1 = u_2 = u_3 = u_4 = 1$; $u_5 = 1 - p^5$ and the following recursion:

$$\begin{aligned} u_{n+5} = & (1-p)u_{n+4} + p(1-p)u_{n+3} + p^2(1-p)u_{n+2} \\ & + p^3(1-p)u_{n+1} + p^4(1-p)u_n \end{aligned}$$

The reliability for Model 2 is $R_N = 1 - u_N$. Curiously enough, separable systems accept an efficient representation, but the reliability evaluation belongs to the class of $\mathcal{NP}$-Hard problems. Therefore, the problem of computing the reliability for Model 3 (for an arbitrary N and an arbitrary random hyperplane) is $\mathcal{NP}$-Hard, since it represents an arbitrary separable system. As exact computation is then computationally hard, an alternative is to employ pointwise reliability estimations for Models 3 and 4 using for instance Crude Monte Carlo method [11]. The reliability evaluation of Model 5 is found by considering the equivalent four-state system composed by the strings $\{00, 01, 10, 11\}$. The reliability is then the fourth component of the vector $\pi' \times Q^{N-2}$, being $\pi' = (\pi_0 M_{(0,0)}, \pi_0 M_{(0,1)}, \pi_1 M_{(1,0)}, \pi_1 M_{(1,1)})$, and Q:

$$Q = \begin{pmatrix} M_{(0,0)} & 1 - M_{(0,0)} & 0 & 0 \\ 0 & 0 & M_{(1,0)} & 1 - M_{(1,0)} \\ M_{(0,0)} & 1 - M_{(0,0)} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix},$$

Finally, the reliability of Model 6 is $R(p) = p^4 + 2p^2(1-p)^2 + 4p^3(1-p)$.

IV. Support Vector Machines

Observe that the number of states of an SBS grows exponentially with respect to the number of components. In practice, the underlying structure of an SBS is only known for partial states. As a consequence, we assume a set of independent samples $\phi(X_1), \ldots, \phi(X_m)$ of the structure, and the goal is to have a computational efficient representation, with low misclassification error.

In order to have a honest estimation of the misclassification error, machine learning methods can be evaluated in a two-step fashion: first, a model is trained/constructed and second, it is tested on a new dataset, which has not contributed to the construction of the model. The reader can consult alternative resampling methods such as cross-validation [12] or bootstrap [13], especially when the number of observations is reduced.

A modern method of statistical learning, Support Vector Machine (SVM), does not impose any assumption on the data distribution. The goal is to find a hyperplane that linearly separates different groups of observations. The rationale behind this construction is to maximize the distance between the two sets, or the margin between the points and the hyperplane. This approach is different from discriminant analysis, whose construction of the separator hyperplane depends on all the data, with hard constraints (i.e., continuous variables and identical covariance matrix for the groups).

Even if the data is not completely linearly separated or if it is actually impossible to find a hyperplane that separates them, SVM, using the *kernel trick*, can map the observations in a space of higher dimension, where it could be much simpler to separate them linearly. The intuition is that it is easier to separate groups in a larger space. The methodology is based on the kernel trick [14], and the fact that an SVM classifier only depends on a dot product. The most used kernels are the radial or gaussian kernel $k_r(x,y) = e^{-\gamma||x-y||^2}$, the linear kernel $k_l(x,y) = \langle x, y\rangle$ and the polynomial kernel $k_p(x,y) = (1+\langle x,y\rangle)^d$ of degree d. The reader is invited to consult [14] or [12] for further details.

V. Results

In this section, we present and discuss results obtained applying the previous ideas to a number of particular test cases, based on the models *(M1)* to *(M6)* defined previously. For each of the models, we considered three values for the number of components, $N \in \{10, 30, 50\}$. For models *(M3)* and *(M4)*, random hyperplanes were sampled to complete the model definitions. In order to apply SV methodology, we used three sample size values, $m \in \{600, 3000, 9000\}$. We now discuss model fitting and then analyze how the results can be used to estimate reliability.

A. Model Fitting

In order to highlight the level of separability and potential representation of an SBS, we considered three different SVM classifiers described in Section IV, to know, linear, radial and polynomial kernels. We carried out the simulations using package `e1071` of `R` software, specifically useful for SVM [15].

As a rule of thumb in machine learning, we considered two-thirds of the sample to build the model (training stage), and the remaining one-third of the sample to validate the model. The performance of the different SVMs is measured in terms of their accuracy, for the six SBSs under study from Section III. Tables I-III report the accuracy of the different SVM classifiers over an averaging of 50 independent-runs, for the different systems under study, considering $N = 10$, $N = 30$ and $N = 50$, respectively.

There is a trade-off between the sampling size m and the accuracy. The corresponding systems for $N \in \{30, 50\}$ have abundant number of states and modest sample size m (see Tables II and III). However, Table I presents a scenario under a *forced overfitting*, with the $2^N = 1024$ states of the system. The matching between the different structures and classifiers is better as m is increased. This fact confirms the consistency of our methods.

	SBS	Linear	Radial	Polynomial
1024	$M1$	1.0000	1.0000	1.0000
	$M2$	0.9334	0.9188	0.9221
	$M3$	0.9677	0.9316	0.9545
	$M4$	0.9452	0.9319	0.9330
	$M6$	0.8930	1.0000	1.0000

TABLE I
Accuracy of the three SVM models with $N = 10$

m	SBS	Linear	Radial	Polynomial
600	$M1$	0.9630	0.8995	0.8566
	$M2$	0.7639	0.7601	0.7742
	$M3$	0.9474	0.9072	0.9048
	$M4$	0.9519	0.9135	0.9249
	$M5$	0.6641	0.6078	0.5530
	$M6$	0.9200	0.8558	0.8129
3000	M1	1.0000	0.9657	0.9733
	M2	0.7476	0.7693	0.7674
	M3	0.9817	0.9596	0.9535
	M4	0.8973	0.9014	0.8982
	M5	0.6621	0.7438	0.7102
	M6	0.9361	0.9240	0.9263
9000	M1	1.0000	0.9860	0.9902
	M2	0.7533	0.8070	0.8006
	M3	0.9918	0.9768	0.9751
	M4	0.9890	0.9781	0.9751
	M5	0.6629	0.8412	0.8233
	M6	0.9431	1.0000	0.9711

TABLE II
Accuracy of the three SVM models with $N = 30$ and $m = 600, 3000, 9000$

The Linear classifier predicts correctly more than 93% of the sample for Models $M1$ and $M3$, for all the cases under study. The matching is even better as the number of components N is increased. This is in correspondence with the fact that $M1$

m	SBS	Linear	Radial	Polynomial
600	M1	0.9488	0.8673	0.8668
	M2	0.7970	0.8215	0.8215
	M3	0.9378	0.8210	0.7851
	M4	0.9147	0.8743	0.8881
	M5	0.6091	0.6372	0.6372
	M6	0.8986	0.8206	0.7814
3000	M1	0.9991	0.9363	0.9125
	M2	0.8273	0.8235	0.8235
	M3	0.9851	0.9255	0.9025
	M4	0.9828	0.9482	0.9420
	M5	0.6471	0.6409	0.6408
	M6	0.9612	0.9607	0.9555
9000	M1	1.0000	0.9695	0.9587
	M2	0.8366	0.8366	0.8366
	M3	0.9905	0.9698	0.9571
	M4	0.9605	0.9557	0.9530
	M5	0.6557	0.6781	0.6483
	M6	0.9680	0.9710	0.9704

TABLE III
ACCURACY OF THE THREE SVM MODELS WITH $N = 50$ AND $m = 600, 3000, 9000$

and $M3$ are separable systems, so actually a perfect solution in theory could attain a 100% accuracy. The reader can appreciate that the method finding an imperfect matching, or sub-optimal hyperplane, is related to the partial knowledge of the structure during the training stage (i.e., not all the states of the structure are known in order to build the model). Furthermore, observe that there is a perfect matching between the Linear classifier and $M1$ using a sample-size of $m = 9000$ for all the cases $N \in \{10, 30, 50\}$. This fact suggests a reasonable sampling-size to consider in real-life separable systems.

The linear classifier has good results for $M4$ (89% or more of the sample predicted correctly). This system is not linearly separable, so that there will always be a discrepancy between a linear separation and the actual system. in this context, the results obtained by the linear classifier seem very good, and they outperform the other two classifiers in most combinations of N and M.

For model $M2$ results are not so clearcut, and in general all clssifiers give relatively poor results (with accuracy 83% in the best case). The results are relatively similar and insensitive on the choice of the classifier, at the same time the polynomial classifier in most cases is the best or obtains results similar to the best for this model.

For model $M5$, the best model overall seems to be the radial one, which in most cases obtains better results than the linear and polynomial one. The accuracies obtained in the $N = 30$ case grow quickly with the sample size m, but in the $N = 50$ case the results are not so promising, with very low success rate (at most 67%).

Regarding model $M6$, for low m values the linear model gas good accuracies, when m increases both the polynomial and the radial model improve, and this latter one reaches very high accuracy cases for both $M = 30$ and $M = 50$ when $m = 9000$.

In an overall perspective, the Linear classifier has better general performance, but the kind of model impacts greatly and for systems which are far away from being linearly separable, the other classifiers may be better options.

B. Reliability Evaluation

Once a classifier is adopted, we can perform a pointwise reliability evaluation, using Crude Monte Carlo (CMC). From a macroscopic viewpoint, Monte Carlo is a noteworthy statistical method to simulate complex systems. The reader can find an extensive analysis of Monte Carlo methods in the authoritative book [11]. Here, the use of CMC is just an averaging among all the sample. Let us denote $\overline{X_n}$ the averaging over the set $X_1, \ldots, X_n$, and consider the three classifiers: $Y_x^i = \mathcal{C}_x(X_i)$, being $x \in \{l, r, p\}$ linear, radial and polynomial estimations. Tables IV-VI show the performance of CMC for the six models treated in the previous section, taking the most complex cases, i.e $N = 50$. Complementarily, Tables VII-IX show the performance of the pointwise reliability evaluation obtained when employing the SVM approximation. The correct reliability value R is analytically found whenever possible as described in Section III (when non-avaible, an NA entry is shown in the tables). An approximate confidence interval $[R_l, R_r]$ with a level of 90% centered at $\overline{\phi(X_m)}$ for the reliability R is also provided. Observe that the reliability evaluation of separable systems is $\mathcal{NP}$-Hard (Theorem 2). Thus, here we do not report the exact reliability evaluation for models $M3$ and $M4$. Observe that both Radial and Polynomial classifiers tend to overestimate the reliability. Furthermore, they tend to classify all the states as operational, specially under small sampling size. The Linear classifier has a small gap with the pointwise unbiased estimation $\overline{\phi(X_m)}$ under the separable models $M3$ and $M4$. Furthermore, the numbers $\overline{\mathcal{C}_l(Y_{m/3})}$ are centered in the corresponding reliability intervals. The reliability estimations are mixed, in some cases the values are part of the confidence intervals, in other ones they are quite afar, suggesting that additional work must be performed to improve the use of the SVM models as a source of reliability estimation.

	$\overline{\phi(X_m)}$	R	R_l	R_r
M1	0.8633	0.8721	0.8377	0.8855
M2	0.8183	0.8350	0.7902	0.8436
M3	0.7900	NA	0.7605	0.8168
M4	0.5283	NA	0.4939	0.5625
M5	0.6367	0.6446	0.6030	0.6691
M6	0.7167	0.7399	0.6846	0.7467

TABLE IV
RELIABILITY ESTIMATION WITH $N = 50$ AND $m = 600$.

	$\overline{\phi(X_m)}$	R	R_l	R_r
M1	0.8623	0.8721	0.8515	0.8725
M2	0.8233	0.8350	0.8114	0.8347
M3	0.9027	NA	0.8932	0.9114
M4	0.2477	NA	0.2348	0.2610
M5	0.6423	0.6446	0.6276	0.6568
M6	0.7437	0.7399	0.7302	0.7567

TABLE V
RELIABILITY ESTIMATION WITH $N = 50$ AND $m = 3000$.

	$\overline{\phi(X_m)}$	R	R_l	R_r
M1	0.8711	0.8721	0.8651	0.8769
M2	0.8373	0.8350	0.8308	0.8437
M3	0.8682	NA	0.8622	0.8740
M4	0.8431	NA	0.8366	0.8494
M5	0.6469	0.6446	0.6385	0.6552
M6	0.7390	0.7399	0.7313	0.7466

TABLE VI
RELIABILITY ESTIMATION WITH $N = 50$ AND $m = 9000$.

Model	$\overline{\mathcal{C}_l(Y_{m/3})}$	$\overline{\mathcal{C}_r(Y_{m/3})}$	$\overline{\mathcal{C}_p(Y_{m/3})}$	R
M1	0.8850	1.0000	1.0000	0.8721
M2	0.8650	1.0000	1.0000	0.8350
M3	0.7750	0.9750	1.0000	NA
M4	0.5150	0.5650	0.5350	NA
M5	0.7850	1.0000	1.0000	0.6446
M6	0.6950	0.8500	0.9250	0.7399

TABLE VII
$N = 50, m = 600$.

Model	$\overline{\mathcal{C}_l(Y_{m/3})}$	$\overline{\mathcal{C}_r(Y_{m/3})}$	$\overline{\mathcal{C}_p(Y_{m/3})}$	R
M1	0.8760	0.9210	0.9450	0.8721
M2	0.9570	1.0000	1.0000	0.8350
M3	0.8980	0.9780	0.9990	?
M4	0.2510	0.2150	0.2060	?
M5	0.9020	1.0000	1.0000	0.6446
M6	0.7060	0.7400	0.7370	0.7399

TABLE VIII
$N = 50, m = 3000$.

Model	$\overline{\mathcal{C}_l(Y_{m/3})}$	$\overline{\mathcal{C}_r(Y_{m/3})}$	$\overline{\mathcal{C}_p(Y_{m/3})}$	R
M1	0.8707	0.8990	0.9090	0.8721
M2	1.0000	1.0000	1.0000	0.8350
M3	0.8677	0.8977	0.9150	NA
M4	0.8507	0.8717	0.8780	NA
M5	0.8930	0.9370	0.9987	0.6446
M6	0.6963	0.6527	0.6527	0.7399

TABLE IX
$N = 50, m = 9000$.

VI. CONCLUSIONS AND TRENDS FOR FUTURE WORK

System reliability has multiple applications, from transportation to the design of communication networks. The reliability analysis of Stochastic Binary Systems (SBS) belongs to the class of $\mathcal{NP}$-Hard problems. Furthermore, special separable SBS accept an efficient representation, but its reliability evaluation remains hard. Prior works in the literature assume independent failures, which is not necessarily a realistic assumption. Also, while it is assumed that the system structure is known perfectly, in practice we can observe the system under a number of particular states or configurations; it is not always possible to observe all its feasible states. Then, it can be a challenge to build a valuable model starting from a finite number of observations or samples.

In this work, we propose a model fitting methodology to build a model of an SBS. Experiments with linear, polynomial and radial SVM were performed. Of course other classifiers could be considered, to see if better fits are posssible. Also, some experiments for giving an estimation of the reliability were performed. The results suggest promising progress for model estimation of separable systems, and more mixed success in the case of non-separable systems, which must be studied with more detail in the future.

Other additional future work can include discussing SBS with dependent failures: independent cluster-failures, cascading failures, two or three-dependent failures. Also, it is an open issue to further develop machine learning techniques in order to build structures which lead to efficient reliability evaluation of general systems.

VII. ACKNOWLEDGEMENTS

This work is partially supported by Project 395 CSIC I+D *Sistemas Binarios Estocásticos Dinámicos*, COST Action 15127 *Resilient communication services protecting end-user applications from disaster-based failures* and Project MATHAMSUD 19-MATH-03 Raredep *Rare events analysis in multi-component systems with dependent components*.

REFERENCES

[1] M. O. Ball and J. S. Provan, "The complexity of counting cuts and of computing the probability that a graph is connected," *SIAM J. Computing*, vol. 12, pp. 777–788, 1983.

[2] A. Rosenthal, "Computing the reliability of complex networks," *SIAM Journal on Applied Mathematics*, vol. 32, no. 2, pp. 384–393, 1977.

[3] C. J. Colbourn, *The Combinatorics of Network Reliability*. New York, NY, USA: Oxford University Press, Inc., 1987.

[4] E. Canale, H. Cancela, F. Robledo, P. Romero, and P. Sartor, "Full complexity analysis of the diameter-constrained reliability," *International Transactions in Operational Research*, vol. 22, no. 5, pp. 811–821, 2015.

[5] H. Cancela, G. Ferreira, G. Guerberoff, F. Robledo, and P. Romero, "Building reliability bounds in stochastic binary systems," in *10th International Workshop on Resilient Networks Design and Modeling (RNDM)*, Aug 2018, pp. 1–7.

[6] M. O. Ball, "Computational complexity of network reliability analysis: An overview," *IEEE Transactions on Reliability*, vol. 35, no. 3, pp. 230 –239, aug. 1986.

[7] J. Conway, *A Course in Functional Analysis*, ser. Graduate Texts in Mathematics. Springer New York, 1994.

[8] M. O. Ball, "Complexity of network reliability computations," *Networks*, vol. 10, no. 2, pp. 153–165, 1980.

[9] L. Petingi and J. Rodriguez, "Reliability of networks with delay constraints," in *Congressus Numerantium*, vol. 152, 2001, pp. 117–123.

[10] S. A. Cook, "The complexity of theorem-proving procedures," in *Proceedings of the third annual ACM symposium on Theory of computing*, ser. STOC '71. New York, NY, USA: ACM, 1971, pp. 151–158.

[11] G. Fishman, *Monte Carlo:*, ser. Springer Series in Operations Research and Financial Engineering. Springer, 1996.

[12] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An Introduction to Statistical Learning: with Applications in R*, ser. Springer Texts in Statistics. Springer New York, 2014.

[13] B. Efron and R. Tibshirani, *An Introduction to the Bootstrap*, ser. Monographs on statistics and applied probability. Chapman & Hall, 1993.

[14] V. N. Vapnik, *The nature of statistical learning theory*. Springer-Verlag New York, Inc., 1995.

[15] D. Meyer, E. Dimitriadou, K. Hornik, A. Weingessel, and F. Leisch, *e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien*, 2017, r package version 1.6-8. [Online]. Available: https://CRAN.R-project.org/package=e1071