Les Cahiers du GERAD ISSN: 0711–2440

Robotic Process Automation (RPA) using a heuristic method and the effective resistance of a graph

H. Tremblay, S. Séguin, L.-A. Boily, V. Du Paul, S. Lalancette

G–2023–12

April 2023

La publication de ces rapports de recherche est rendue possible grâce au soutien de HEC Montréal, Polytechnique Montréal, Université McGill, Université du Québec à Montréal, ainsi que du Fonds de recherche du Québec – Nature et technologies.

Dépôt légal – Bibliothèque et Archives nationales du Québec, 2023
– Bibliothèque et Archives Canada, 2023

The publication of these research reports is made possible thanks to the support of HEC Montréal, Polytechnique Montréal, McGill University, Université du Québec à Montréal, as well as the Fonds de recherche du Québec – Nature et technologies.

Legal deposit – Bibliothèque et Archives nationales du Québec, 2023
– Library and Archives Canada, 2023

GERAD HEC Montréal
3000, chemin de la Côte-Sainte-Catherine
Montréal (Québec) Canada H3T 2A7

Tél. : 514 340-6053
Téléc. : 514 340-5665
info@gerad.ca
www.gerad.ca

Robotic Process Automation (RPA) using a heuristic method and the effective resistance of a graph

Hugo Tremblay [a]

Sara Séguin [a, b]

Laurie-Ann Boily [a]

Véronique Du Paul [a]

Sophie Lalancette [a]

[a] *Université du Québec à Chicoutimi, Saguenay (Qc), Canada, G7H 2B1*

[b] *GERAD, Montréal (Qc), Canada, H3T 1J4*

h7trembl@uqac.ca
sara.seguin@uqac.ca

April 2023
Les Cahiers du GERAD
G–2023–12

Abstract : Robotic Process Automation has emerged in recent years as an important field by allowing faster and more secure processes through a reduction in the risks or errors but also an increase in the productivity rates of many industries. In this specific paper, the RPA problem aims at assigning financial transactions to software robots to minimize the total costs, incured by the licenses and utilization time. The problem is represented as a bipartite graph and the effective resistance of the graph, which is analog to an electrical circuit, is used to order the edges of a heuristic method to assign the transactions to robots. Preliminary results, based on real data from a bank, are compared to the optimal solution obtained by a linear integer programming model. They show that the heuristic method allows to obtain results quicker and that they are near the optimal solution.

Keywords: Robotic process automation, effective resistance, graph theory, heuristic, assignment problem

---

Acknowledgements: The authors would like to thank the Natural Sciences and Engineering Research Council of Canada (NSERC) for providing funds to conduct this research.

1 Introduction

Robotic Process Automation gained a lot of attention in the past five years, both in the fields of academia and industry. The interest for RPA started in industry, as companies always want to increase profit by reducing their operations and also possible errors, since treating high volumes of operations necessary leads to some imperfections. The interest from academia aroused afterwards and in the past five years, many papers and literature reviews in the field have been published [2, 4, 5, 9, 11, 14].

The goal of RPA is to mimic human behaviour in accomplishing repetitive tasks. Software is programmed to replicate the operations normally done by a person, which decreases the risk of errors, and increases the number of tasks which in turn increases productivity. More specifically, banks have millions of operations to treat every day, and many of them are repetitive. For example, transferring funds in between accounts requires a person to click on the screen and enter the amount. This can easily automated by RPA, as the software is programmed to do the same tasks, but quicker and with a low risk of error. An interesting paper [12] specifically studied the banking sector, and provides pros and cons of RPA applied to banks. Many papers provide case studies applied to various fields. For example, in 2015, the company Telfonica O2 [8] had around 450,000 automated tasks with RPA each month and with this success they are planning in integrating more automation. The authors of [10] state that the recruitment process for companies would be a lot more efficient if algorithms were used to help the recruiters make a decision faster.

Many papers deal with the integration of RPA in the business process management (BPM) of companies [1, 7], but few papers are concerned with the actual algorithms that are launched in the back-end to accomplish repetitive tasks.

The field of operations research is concerned with maximizing profit of minimizing costs of real-world problems, which easily applies to RPA. In this paper, the automation of tasks for a financial institution is presented. The problem consists in completing a certain volume of tasks of different type, at the lowest possible cost, considering that a software robot has a licence cost. Each type of tasks requires a certain time to be completed and market opening hours need to be respected when completing the tasks, since the problem is applied to the banking sector. This problem is NP-Hard, and an attempt to solve this problem using a linear integer programming model was proposed in [15]. Four heuristic methods were tested to compute an upper bound on the number of robots required, which is then given as an input to a linear integer program to determine the assignation of the transactions and their volumes to the different robots. Solutions are obtained quickly, but the problem is deterministic, which means that the number and volume of transactions is known beforehand. In practice, these values change during the day and a dynamic algorithm would allow to obtain better results.

Graph theory allows to model problems with nodes and edges. In the case of this specific problem, the nodes are the types of transactions and the periods, whereas the edges between these nodes exist if a transaction type can be completed at a certain period. The advantage of modeling the problem as a graph is the flexibility of adding transaction types, periods and volumes, therefore providing a more dynamic solution. As for the costs on the edges, a novel method, the effective resistance of a graph [3, 6], is chosen to compute their values. The effective resistance is a metric that is analog to an electrical circuit. The idea behind this choice is that effective resistance is transparent to the problem, therefore, the idea can be generalized to any application.

The novelty of the problem presented in this paper is twofolds. First, the formulation of the problem as a graph allows a dynamic setting and second, the use of effective resistance to obtain values on the edges on an RPA application is new. Numerical results show that the number or robots required to minimize their number and complete all of the transactions is similar to the exact solution obtained with the integer linear program [15], although the proposed methodology is an heuristic.

The paper is organized as follows. Section 2 presents the RPA assignment problem. Section 3 explains the methodologye, more specifically the effective resistance metric for the edge costs and the proposed heuristic for the assignment of transactions to periods. Numerical results are presented in Section 4 and concluding remarks in Section 5.

2 Mathematical model

We model the RPA problem using a weighted graph: Let $G = (P \cup K, E, w)$ be a weighted bipartite graph where the edge (p, k) is in E if and only if transaction p can be processed at period k and $w_{p,k}$ is the weight of edge (p, k). Then, a set of weighted edges of G is a valid assignation of the transactions using a single robot if and only if $\forall p \in P$, $\sum_{(p,k_i)} w_{p,k_i} = v_p$ and $\forall k \in K$, $l_k - \sum_{(p_i,k)} w_{p_i,k} \cdot t_{p_i} \geq 0$. In other words, an assignation is valid for a single robot if all transactions are processed without exceeding the length of any period. Figure 1 shows the graph of a valid assignation for a small instance of three transactions to four periods.

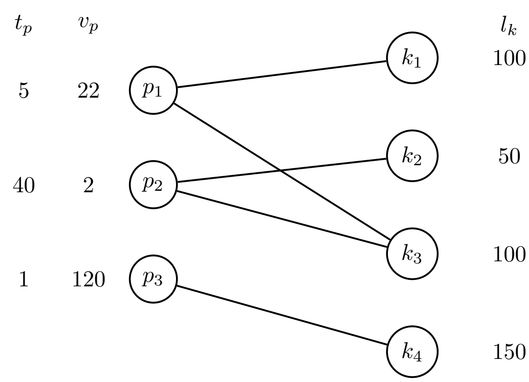

Figure 1: Graph model of an RPA problem with 3 transactions and 4 periods. The values t_p, v_p and l_k are indicated next to each relevant node.

The operation of adding a new robot augments the graph G by adding a disjoint copy K' of the set of periods and adding the edge (p_i, k'_j) whenever the edge $(p_i, k'j)$ exists in G. So, by iteratively adding new robots until a valid assignation is found, one obtains the solution to the RPA problem.

3 Heuristics to compute a valid assignation

By iteratively adding new robots until a valid assignation is found, one obtains the solution to the RPA problem. However, checking for optimality at each step is NP-complete [13]. We thus devise an heuristic to obtain a good valid assignation quickly. It is based on the effective resistance of a graph.

3.1 Effective resistance of a graph

As its name suggest, the effective resistance of a graph is based on the notion of effective resistance of an electrical circuit. A broader overview of the subject can be found in [3, 6]. Given a weighted undirected graph $G = (V, E, w)$ of order n, each edge $e = (a, b) \in E$ is treated as a resistor with resistance $r(e) = 1/w_e$. The reason for taking the inverse of w_e is that an edge with a very small weight is very restrictive and should be used with a high priority. Then, the effective resistance R_{ab} between vertices a and b is the total resistance of the electrical circuit obtained by connecting an electrical source across a and b. Intuitively, the effective resistance between a and b gives and idea of how difficult it is to get from vertex a to b in G, considering the whole structure of G as opposed to the single edge (a, b).

When G is relatively small, R_{ab} is easily computed using Kirchhoff's circuit laws. For larger graphs however, we use notions of spectral graph theory in order to compute the effective resistance between all pairs of vertices: Let A and $D \in Mat_n(\mathbb{R})$ be the adjacency matrix and degree matrix of G respectively. Then, the matrix $L = D - A$ is called the Laplacian of G. Now, set $1_n^\perp = span\{1_n\}^\perp$, the subspace of $\mathbb{R}^n$ perpendicular to $1_n = (1, 1, \ldots, 1)$ and Q a matrix whose lines form a basis of $1_n^\perp$. By considering the reduced Laplacian $\overline{L} = QLQ^T$ and solving the Lyapunov equation:

$$\overline{L}\Sigma + \Sigma\overline{L}^T = I_{n-1}, \tag{1}$$

effective resistance of each pair of vertices is given by:

$$R_{kj} = x_{kk} + x_{jj} - 2x_{kj}, \tag{2}$$

where $X = 2Q^T\Sigma Q$.

It is worth noticing that the effective resistance is a graph distance for undirected graphs. Also, tt is indeed possible to define R_{ab} for directed graphs but this is beyond the scope of this paper.

3.2 Heuristics and implementation

Using the graph model of Section 2 and the effective resistance, we tested the following heuristics for computing a valid assignation for the RPA problem:

1. Compute the effective resistance R_pk for all edges (p, k);
2. Sort the edges by decreasing effective resistance value;
3. Construct an assignation by loading each successive edge with the maximum available volume;
4. Check for the validity of the assignation;
5. If it is valid, end the algorithm. Otherwise, add a new robot (by adding a new copy of K) and go back to step 1.

This process was implemented in the python language using the numpy package. For example, Table 1 and Figure 2 respectively show the effective resistance and the resulting valid assignation of the graph from Figure 1. An effective resistance value of infinity denotes that no path exist between the vertices. Remark that, although edges (p_1, k_2) and (p_2, k_1) do not exist in G, their respective resistance is not infinity since from a graph theoretic perpective, there exists en undirected path in G between those vertices. It is also worth mentioning that in this example, our method returns a valid assignation using a single robot, which is of course optimal.

Table 1: Effective resistance for the edges of graph G as computed using the method presented in Section 3.1. Values in parentheses are irrelevant for our heuristics since the associated edges are not present in the graph. This induces the following ordering of the edges: (p_3, k_4), (p_2, k_3), (p_1, k_3), (p_2, k_2), (p_1, k_1).

	k_1	k_2	k_3	k_4
p_1	0.9375	(2.9375)	1.0000	(∞)
p_2	(3.0000)	1.0000	1.0625	(∞)
p_3	(∞)	(∞)	(∞)	1.2500

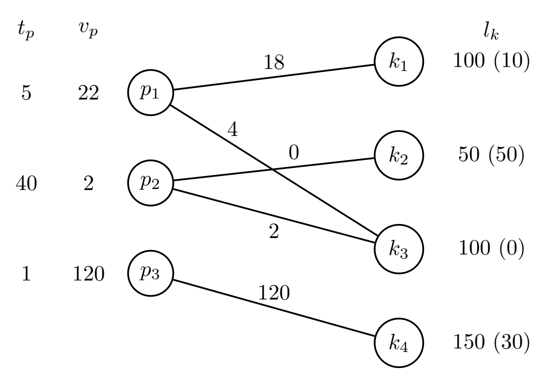

Figure 2: Valid assignation obtained from the effective resistance of Table 1. l_k values in parentheses show the unused time for each period.

4 Experimental results

Results for this study are conducted on a single test case of real data provided by a financial institution. As this is a preliminary project, the goal of the results is to compare the solution obtained with a linear integer problem, that is optimal, and the heuristic proposed in this paper, to show that using the effective resistance to order the edges of the graph for the assignement heuristic is effective and can be investigated further.

The data used consists of 12 different transaction types, that each have a different treatment time, volume, market opening hours and clearance during which the transactions can be processed and must be treated. Table 2 presents the numerical values used in the test case.

Table 2: Data for the test case. Transaction types with their duration, volume, and market opening hours.

Type	1	2	3	4	5	6	7	8	9	10	11	12
Time begin (h)	7	8	5,75	7	6	7	8	7	7	7	7	7
Time end(h)	21	17	7	23	23	19	20	19	19	23	21	19
Duration (s)	120	30	20	15	90	360	60	50	1320	150	480	52
Volume	220	772	509	13750	450	210	110	375	95	150	235	76

This section present the results obtained using the linear integer program, and the proposed heuristic in this study.

4.1 Optimal solution

The RPA problem was first solved using an integer linear program, as presented in [13]. As the goal of this paper is to determine if the proposed heuristic allows to obtain a good solution, we refer the readers to the paper to obtain all of the mathematical details related to the optimal solution, and simply report the solution here. One of the main drawbacks of the linear integer program is that the RPA problem is NP-hard. Therefore, for few transaction types, the problem can be easily solved, but when this number increases, the problem is difficult to solve. The solution computed by the linear integer program is optimal, which means that it is the best possible solution for the RPA test case.

The optimal solution is shown in Table 3. A total of 11 robots and 9 periods are required to process all of the transaction types.

4.2 Heuristic solution

Results for the heuristic proposed in this paper are reported in Table 4. The solution requires 12 robots and 8 periods to process all of the transactions.

Results show that the solution is close to the optimal solution, thus requiring an additional robot but fewer number of periods. In practice, this requires additional costs, since each robot has a licence cost, as well as operation costs related to the time they are actually used. The comparison between optimal results and heuritic results show that in the optimal solution, there are more periods during which some robots are inactive, leading to less usage time.

For this specific test case, the cost are higher than the optimal solution, but the results also show that the solution is quite viable. In fact, obtaining the solution with heuristics is faster than with the linear integer problem, and the results are near the optimal solution.

Table 3: Optimal results from linear integer program. Transaction type numbers are given under the robot column and number of processed transactions are in ().

Period	Robot 1	Robot 2	Robot 3	Robot 4	Robot 5	Robot 6
1	3(225)	3(225)	3(59)	-	-	-
2	5(40)	5(40)	5(40)	5(40)	5(16)	-
3	1(30)	1(30)	1(30)	1(30)	1(30)	1(10),12(7)
4	9(17),11(2)	9(17),11(2)	8(7),11(48)	9(10),11(21)	8(7),11(48)	9(17),11(2)
5	6(479)	6(40),10(110)	6(480)	6(480)	6(480)	6(480)
6	4(479)	4(40),7(110)	4(480)	4(480)	4(480)	4(480)
7	4(480)	4(480)	4(480)	4(480)	4(480)	4(480)
8	4(480)	4(480)	4(480)	4(480)	4(480)	4(480)
9	2(770)	-	-	-	-	-
Period	Robot 7	Robot 8	Robot 9	Robot 10	Robot 11	Robot 12
1	-	-	-	-	-	-
2	-	-	-	-	-	-
3	1(30)	1(30)	12(69)	-	-	-
4	8(199),9(10)	9(17),11(2)	8(7),11(48)	8(4),9(7),11(29)	8(151),11(33)	-
5	6(480)	6(480)	6(480)	6(480)	6(480)	-
6	4(480)	4(480)	4(480)	4(480)	4(480)	-
7	5(80)	4(272),5(34)	5(80)	4(480)	5(80)	-
8	4(480)	4(480)	4(480)	4(480)	4(480)	-
9	-	-	-	-	-	-

Table 4: Heuristic results. Transaction type numbers are given under the robot column and number of processed transactions are in ().

Period	Robot 1	Robot 2	Robot 3	Robot 4	Robot 5	Robot 6
1	3(45)	3(45)	3(45)	3(14)	-	-
2	5(40)	3(180)	3(180)	5(40)	5(40)	5(40)
3	8(72)	6(10)	6(10)	12(69)	9(2),11(2)	1(8),9(2)
4	2(772),4(2),8(184)	6(90)	6(79),9(3)	1(2),9(24),11(1)	1(2),9(24),11(1)	1(39),9(21)
5	7(50),8(84)	6(20)	4(2),6(1),8(4),9(5)	8(31),9(4),12(7)	1(1),9(5),11(1)	1(5),9(5)
6	7(60)	1(2),11(7)	1(30)	1(2),11(7)	1(2),11(7)	1(30)
7	1(2),11(7)	1(2),11(7)	1(30)	1(2),11(7)	1(2),11(7)	1(30)
8	4(480)	10(48)	10(48)	4(480)	5(80)	5(80)
Period	Robot 7	Robot 8	Robot 9	Robot 10	Robot 11	Robot 12
1	-	-	-	-	-	-
2	5(40)	5(40)	-	-	-	-
3	1(2),11(7)	4(16),11(7)	4(240)	4(240)	4(240)	-
4	1(2),11(67)	4(656),11(47)	4(2160)	4(2160)	4(2160)	-
5	11(15)	11(15)	4(480)	4(480)	4(480)	-
6	1(2),11(7)	1(19),4(24),11(2)	4(240)	4(240)	4(240)	-
7	1(2),11(7)	1(2),11(7)	4(240)	4(240)	4(240)	-
8	10(48)	4(120),5(50),10(6)	4(480)	4(480)	4(480)	4(450)

4.3 Discussion

As mentioned in the beginning of the Section, these results are preliminary. The optimal solution for the RPA is known, but one of the main problems with the linear integer problem is that the problem is difficult to solve when the number of transactions increase. For this test case, the number of transactions is only 12, therefore obtaining the optimal solution is not too difficult. In a real operational setting, the number of transaction types is dynamic, and their volume also change during the day. The advantage of using a graph is that the problem can be easily modified, increased, and changed as the transaction types and their volume change. The advantage of using the effective resistance to order the edges of the graph for the assignment is that the values are unrelated to the actual parameters of the problem, such as the volume, the processing times and so forth, only to the structure of the graph. If the problem changes, the graph changes and it is easy to recompute the effective resistance in order to complete the new assignement, as it only depends on the nodes and edges of the graph. Also, using the effective resistance is novel in the field and can be generalized to any assignment problem. Preliminary results show that the proposed heuristic is a good choice to solve the problem quickly in order to obtain a solution that is close to optimality. Of course, other numerical results need to be conducted to verify this hypothesis, but the results reported in this paper are promising.

5 Conclusion

This paper proposes a novel heuristic that uses the effective resistance of a graph to assign financial transactions to software robots, allowing to solve a RPA problem. The effective resistance is a metric based on the structure of a graph and is analog to an electrical circuit. Preliminary results are conducted on a test case of real data provided by a bank. The solution computed with the proposed heuristic is compared to the optimal solution, obtained from a linear integer problem, and shows that preliminary results are promising. As the RPA problem is NP-hard, the heuristic also allows to obtain results faster. The effective resistance metric is novel in the field and is easily adaptable to any assignment problem, as it is calculated on the properties of the graph, and not the specific problem parameters. Future work based on these promising results are twofolds. First, the formal analysis of the effective resistance of a graph will be investigated thoroughly. Second, many improvements can be tested on the proposed heuristic, for example, the ordering of the edges when completing the assignement, or their weights.

References

[1] S. Aguirre and A. Rodriguez. Automation of a business process using robotic process automation (RPA). In Applied Computer Sciences in Engineering, pages 65–71. Springer International Publishing, 2017.

[2] Wasique Ali Ansari, Paritosh Diya, Sahishnu Patil, and Sunita Patil. A review on robotic process automation-the future of business organizations. In 2nd International Conference on Advances in Science & Technology (ICAST), 2019.

[3] W. Ellens, F.M. Spieksma, P. Van Mieghem, A. Jamakovic, and R.E. Kooij. Effective graph resistance. Linear Algebra and its Applications, 435(10):2491–2506, 2011.

[4] José Gonzalez Enríquez, Andres Jiménez-Ramírez, Francisco José Domínguez-Mayo, and Julián Alberto García-García. Robotic process automation: a scientific and industrial systematic mapping study. IEEE Access, 8:39113–39129, 2020.

[5] L. Ivančić, D. Suša Vugec, and V. B. Vukšić. Robotic process automation: Systematic literature review. In International Conference on Business Process Management, pages 280–295. Springer, 2019.

[6] D. J. Klein and M. Randić. Resistance distance. Journal of Mathematical Chemistry, 12:81–95, 1993.

[7] Maximilian König, Leon Bein, Adriatik Nikaj, and Mathias Weske. Integrating robotic process automation into business process management. In Business Process Management: Blockchain and Robotic Process Automation Forum: BPM 2020 Blockchain and RPA Forum, Seville, Spain, September 13–18, 2020, Proceedings 18, pages 132–146. Springer, 2020.

[8] M. Lacity, L. Willcocks, and A. Craig. Robotic process automation at telefonica o2. Technical report, London, US: University of Missouri-St-Louis — The London School of Economics and Political Science, 2015.

[9] Sílvia Moreira, Henrique S Mamede, and Arnaldo Santos. Process automation using rpa–a literature review. Procedia Computer Science, 219:244–254, 2023.

[10] Dr Nishad Nawaz. Robotic process automation for recruitment process. International Journal of Advanced Research in Engineering and Technology, 10(2), 2019.

[11] Jorge Ribeiro, Rui Lima, Tiago Eckhardt, and Sara Paiva. Robotic process automation and artificial intelligence in industry 4.0–a literature review. Procedia Computer Science, 181:51–58, 2021.

[12] M. Romao, J. Costa, and C. J. Costa. Robotic process automation: A case study in the banking industry. In 2019 14th Iberian Conference on Information Systems and Technologies (CISTI), pages 1–6, 2019.

[13] S. Séguin and I. Benkalaï. Robotic process automation (RPA) using an integer linear programming formulation. Cybernetics and systems, 51(4):357–369, 2020.

[14] R. Syed, S. Suriadi, M. Adams, W.Bandara, S. J.J. Leemans, C. Ouyang, A. H.M. ter Hofstede, I. van de Weerd, M. Thandar Wynn, and H. A. Reijers. Robotic process automation: Contemporary themes and challenges. Computers in Industry, 115:103162, 2020.

[15] Sara Séguin, Hugo Tremblay, Imène Benkalaï, David-Emmanuel Perron-Chouinard, and Xavier Lebeuf. Minimizing the number of robots required for a robotic process automation (RPA) problem. Procedia Computer Science, 192:2689–2698, 2021. Knowledge-Based and Intelligent Information & Engineering Systems: Proceedings of the 25th International Conference KES2021.