



Lot Sizing dengan *Evolutionary Algorithm* dan Excel Solver

Muhammad Ardhya Bisma✉

Program Studi D4 Logistik Bisnis, Sekolah Vokasi, Universitas Logistik dan Bisnis Internasional, Bandung

DOI: 10.31004/jutin.v8i1.42506

✉ Corresponding author:
[bisma@ulbi.ac.id]

Article Info

Abstrak

Kata kunci:

Lot Sizing;

Heuristic;

Metaheuristic;

Evolutionary Algorithm;

Excel Solver

Lot Sizing merupakan proses penentuan jumlah pesan produk yang optimal untuk setiap periode dengan mempertimbangkan beberapa faktor, antara lain ketidakpastian *demand*, *fixed costs*, *quantity discount* dan ongkos simpan, dengan tujuan meminimalkan total biaya. *Evolutionary Algorithm (EA)* merupakan sebuah *metaheuristic* yang dapat diaplikasikan pada berbagai permasalahan optimasi, termasuk *Lot Sizing*. *Solver* merupakan *Excel Add-In* untuk menyelesaikan permasalahan optimasi. Dengan adanya kemampuan *Evolutionary Algorithm* untuk menyelesaikan *Lot Sizing* dan *Solver* yang mampu menjalankan *Evolutionary Algorithm*, maka dapat disimpulkan bahwa penyelesaian *Lot Sizing* dapat dilakukan dengan memanfaatkan *Solver*. Untuk contoh kasus pada penelitian ini, *Lot Sizing* yang dihasilkan dari *Evolutionary Algorithm* dan *Solver* memerlukan *Total Cost* sebesar 62.090 Rubel.

Abstract

Keywords:

Lot Sizing;

Heuristic;

Metaheuristic;

Evolutionary Algorithm;

Excel Solver

Lot Sizing is the process of determining the optimal number of product orders for each period by considering several factors, including demand uncertainty, fixed costs, quantity discounts and holding costs, with the aim of minimizing total costs. *Evolutionary Algorithm (EA)* is a metaheuristic that can be applied to various optimization problems, including *Lot Sizing*. *Solver* is an Excel Add-In for solving optimization problems. With the ability of *Evolutionary Algorithm* to tackle *Lot Sizing* and the *Solver* which is capable of running *Evolutionary Algorithm*, it can be concluded that solving *Lot Sizing* can be done using the *Solver*. For the case example in this research, *Lot Sizing* produced from the *Evolutionary Algorithm* and *Solver* requires a total cost of 62.090 Rubles.

1. INTRODUCTION

Lot Sizing merupakan proses penentuan jumlah pesan produk yang optimal untuk setiap periode dengan mempertimbangkan beberapa faktor, antara lain ketidakpastian *demand*, *fixed costs*, *quantity discount* dan ongkos simpan, dengan tujuan meminimalkan total biaya (Jiao et al., 2017). *Lot Sizing* merupakan *NP-Hard Problem* (Akbalik et al., 2017). *NP-Hard Problem* merupakan sebuah permasalahan dimana untuk memperoleh solusi terbaiknya diperlukan waktu perhitungan yang tidak wajar untuk ditunggu (Mor et al., 2021). Algoritma yang bekerja berdasarkan *heuristic* dan *metaheuristic* dapat menjadi sebuah solusi untuk menyelesaikan *NP-Hard Problem* dengan waktu perhitungan yang wajar untuk ditunggu, walaupun solusi yang dihasilkan belum tentu merupakan solusi yang terbaik (Agrawal et al., 2021).

Beberapa *heuristic* yang populer dalam menyelesaikan *Lot Sizing* adalah sebagai berikut ini (Shenoy & Rosas, 2018): *Lot-For-Lot (LFL)*, *Part-Period Balancing (PPB)*, *Least Unit Cost (LUC)*, *Silver Meal (SM)*, *Wagner-Whitin (WW)*, *Periodic Order Quantity (POQ)*.

Beberapa *metaheuristic* yang populer dalam menyelesaikan *Lot Sizing* adalah sebagai berikut ini (Dorronsoro et al., 2022): *Evolutionary Algorithm (EA)*, *Genetic Algorithm (GA)*, *Ant Colony Optimization (ACO)*, *Particle Swarm Optimization (PSO)*, *Tabu Search (TS)*, *Simulated Annealing (SA)*.

Evolutionary Algorithm (EA) merupakan sebuah *metaheuristic* yang dapat diaplikasikan pada berbagai permasalahan optimasi, termasuk *Lot Sizing*. *Evolutionary Algorithm* merupakan sebuah *metaheuristic* yang terinspirasi dari teori evolusi, dimana individu yang terkuat yang akan bertahan hidup (Badar, 2021).

Solver merupakan *Excel Add-In* untuk menyelesaikan permasalahan optimasi, yang bertugas untuk menemukan nilai-nilai untuk sekumpulan variabel keputusan dengan tujuan memaksimalkan dan meminimumkan tujuan (Selvi et al., 2022). *Solver* menggunakan beberapa algoritma untuk menemukan nilai optimal yang akan digunakan pada variabel-variabel keputusan. Tiga algoritma tersebut adalah sebagai berikut ini (Gilbert et al., 2023): *Linear Programming Simplex*, *GRG Nonlinear*, *Evolutionary Algorithm*.

Dengan adanya kemampuan *Evolutionary Algorithm* untuk menyelesaikan *Lot Sizing* dan *Solver* yang mampu menjalankan *Evolutionary Algorithm*, maka dapat disimpulkan bahwa penyelesaian *Lot Sizing* dapat dilakukan dengan memanfaatkan *Solver*.

Kualitas hasil perhitungan *Lot Sizing* dengan *Evolutionary Algorithm* dan *Solver* perlu diukur, salah satu cara mengukurnya adalah dengan membandingkannya dengan hasil perhitungan *Lot Sizing* berdasarkan algoritma yang lain, pada *paper* kali ini akan dibandingkan dengan algoritma *Lot-For-Lot*.

2. METHODS

Langkah-langkah dalam penelitian ini adalah sebagai berikut ini:

1. Penentuan Contoh Kasus *Lot Sizing*
Akan diberikan sebuah contoh kasus *Lot Sizing* yang akan diselesaikan.
2. Menyiapkan *Template Excel* Untuk Menghitung *Lot Sizing*
Akan disiapkan sebuah *Template Excel* yang dapat membantu proses perhitungan *Lot Sizing*. Di dalam *Template Excel* ini akan terdapat sebuah tabel yang dilengkapi dengan *formula*-nya.
3. Menghitung *Lot Sizing* berdasarkan algoritma *Lot-For-Lot*
Akan dihitung *Lot Sizing* untuk setiap periodenya berdasarkan algoritma *Lot-For-Lot* dan akan dihitung pula *Total Cost* yang dihasilkannya.
4. Menghitung *Lot Sizing* berdasarkan *Evolutionary Algorithm & Solver*
Akan dihitung *Lot Sizing* untuk setiap periodenya berdasarkan *Evolutionary Algorithm & Solver* dan akan dihitung pula *Total Cost* yang dihasilkannya.
5. Menganalisis Kualitas Hasil Perhitungan
Akan dianalisis kualitas *Lot Sizing* berdasarkan *Evolutionary Algorithm & Solver*, dengan cara membandingkannya dengan *Lot Sizing* berdasarkan algoritma *Lot-For-Lot*.

3. RESULT AND DISCUSSION

Berikut ini merupakan hasil penelitian berdasarkan langkah-langkah pada bagian *methods*.

A. Contoh Kasus *Lot Sizing*

Contoh kasus *Lot Sizing* diperlukan untuk menguji algoritma yang akan digunakan. Berikut ini merupakan data dari contoh kasus yang akan digunakan.

Product Name : Azul
 Product Code : AZU
 Product Cost : 1000 Rubel/Unit
 Ordering Cost : 500 Rubel/Unit
 Holding Cost : 1%
 On-Hand : 14 Unit
 Safety Stock : 5 Unit
 Lead Time : 0 Period
 Lot Size : 1 Unit

Period	1	2	3	4	5	6	7	8	9	10	11	12
GR	4	9	4	6	8	10	6	7	8	1	8	5
SR	5	0	0	3	0	0	0	0	0	0	0	0

GR : Gross Requirement

SR : Scheduled Receipts

B. Template Excel Untuk Menghitung Lot Sizing

Berikut ini merupakan *Template Excel* untuk menghitung *Lot Sizing*:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Pro. No			Quan.			Hol. Cos.			LT (Period)		R Tot. (Unit)				
2	Pro. Par.			BU			OH (Unit)			LS (Unit)						
3	Pro. Nam.			Pro. Cos. (Rub/Unit)			SS (Unit)									
4	Pro. Cod.			Ord. Cos. (Rub/Unit)			OP									
5																
6	Period	11	12	1	2	3	4	5	6	7	8	9	10	11	12	
7	GR I															
8	SS Req.															
9	SR															
10	GR II															
11	PAB I															
12	GR III															
13	PAB II															
14	GR IV															
15	NR															
16	PO Rec.															
17	PO Rel.															
18																
19	PAB III															
20	PAB IV															
21																Cost (Rub)
22	Pro. Cos. (Rub)															
23	Ord. Cos. (Rub)															
24	Hol. Cos. (Rub)															
25																Total Cost
26																

Gambar 1: Template Excel Untuk Menghitung Lot Sizing

Formula pada Template Excel:

- GR I (Gross Requirement I) – Period 1 sd 12
(diisi sesuai data)
- SS Req. (Safety Stock Requirement) – Period 1 sd 12
=IF(D6=1;\$H\$3;0)
- SS (Scheduled Receipts)
(diisi sesuai data)
- GR II (Gross Requirement II) – Period 1 sd 12
=IF((C11+D9)>=(D7+D8);0;(D7+D8)-(C11+D9))
- PAB I (Project Available Balance I) – Period 12 Tahun Sebelumnya
=H2
- PAB I (Project Available Balance I) – Period 1 sd 12
=IF((C11+D9)<(D7+D8);0;(C11+D9)-(D7+D8))
- GR III (Gross Requirement III) – Period 1 sd 12
=IF(C20+D9>=(D7+D8);0;(D7+D8)-(C20+D9))
- PAB II (Project Available Balance II) – Period 12 Tahun Sebelumnya
=H2
- PAB II (Project Available Balance II) – Period 1 sd 12
=C20+D9
- GR IV (Gross Requirement IV) – Period 1 sd 12
(diisi sesuai algoritma yang digunakan)
- NR (Net Requirement) – Period 1 sd 12

- =D14
- *PO Rec. (Planned Order Receipts) – Period 1 sd 12*
=ROUNDUP((D15/\$K\$2);0)*\$K\$2
 - *PO Rel. (Planned Order Release) – Period 11 Tahun Sebelumnya sd 12*
=IFERROR(INDEX(\$B\$16:\$O\$16;1;COLUMN()-1+\$K\$1);0)
 - *PAB III (Project Available Balance III) – Period 11 Tahun Sebelumnya sd 12*
=B13+B16
 - *PAB IV (Project Available Balance IV) – Period 11 Tahun Sebelumnya sd 12*
=B19-(B7+B8)
 - *Pro. Cos. (Product Cost) – Period 11 Tahun Sebelumnya sd 12*
=B15*\$E\$3
 - *Ord. Cos. (Ordering Cost) – Period 11 Tahun Sebelumnya sd 12*
=IF(B17>0;\$E\$4;0)
 - *Hol. Cos. (Holding Cost) – Period 11 Tahun Sebelumnya sd 12 Tahun Sebelumnya*
=B20*\$E\$3*\$H\$1
 - *Hol. Cos. (Holding Cost) – Period 1 sd 12*
=(D20*\$E\$3*\$H\$1)+(\$H\$3*\$E\$3*\$H\$1)
 - *Cost*
=SUM(B22:O22)
 - *Total Cost*
=SUM(P22:P24)
 - *R Tot. (R Total)*
=SUM(D10:O10)

Bagi pembaca yang membutuhkan *file* dari *Template Excel* ini dapat langsung menghubungi penulis melalui surat elektronik.

C. Lot Sizing Berdasarkan Algoritma Lot-For-Lot

Lot-For-Lot merupakan sebuah algoritma yang menentukan jumlah produk yang dipesan atau diproduksi sama dengan jumlah kebutuhan pada periode tertentu, *Lot-For-Lot* ini akan meminimalkan *Holding Cost* tetapi akan meningkatkan *Ordering Cost* (Lu, 2022).

Pada gambar 2 dapat dilihat hasil perhitungan *Lot Sizing* berdasarkan Algoritma *Lot-For-Lot*.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1	Pro. No	1	Quan.	1			Hol. Cos.	1%		LT (Period)	0		R Tot. (Unit)	59		
2	Pro. Par.	0	BU	Each			OH (Unit)	14		LS (Unit)	1					
3	Pro. Nam.	Azul	Pro. Cos. (Rub/Unit)	1000			SS (Unit)	5								
4	Pro. Cod.	AZU	Ord. Cos. (Rub/Unit)	500			OP	Lot For Lot								
5																
6	Period	11	12	1	2	3	4	5	6	7	8	9	10	11	12	
7	GR I			4	9	4	6	8	10	6	7	8	1	8	5	
8	SS Req.			5	0	0	0	0	0	0	0	0	0	0	0	
9	SR			5	0	0	3	0	0	0	0	0	0	0	0	
10	GR II			0	0	3	3	8	10	6	7	8	1	8	5	
11	PAB I		14	10	1	0	0	0	0	0	0	0	0	0	0	
12	GR III			0	0	3	3	8	10	6	7	8	1	8	5	
13	PAB II		14	19	10	1	3	0	0	0	0	0	0	0	0	
14	GR IV			0	0	3	3	8	10	6	7	8	1	8	5	
15	NR			0	0	3	3	8	10	6	7	8	1	8	5	
16	PO Rec.			0	0	3	3	8	10	6	7	8	1	8	5	
17	PO Rel.	0	0	0	0	3	3	8	10	6	7	8	1	8	5	
18																
19	PAB III	0	14	19	10	4	6	8	10	6	7	8	1	8	5	
20	PAB IV	0	14	10	1	0	0	0	0	0	0	0	0	0	0	
21																
22	Pro. Cos. (Rub)	0	0	0	0	3000	3000	8000	10000	6000	7000	8000	1000	8000	5000	59000
23	Ord. Cos. (Rub)	0	0	0	0	500	500	500	500	500	500	500	500	500	500	5000
24	Hol. Cos. (Rub)	0	140	150	60	50	50	50	50	50	50	50	50	50	50	850
25																
																Total Cost
																64850

Gambar 2: Hasil Perhitungan *Lot Sizing* Berdasarkan Algoritma *Lot-For-Lot*

Kebutuhan setiap periode setelah memperhitungkan *Scheduled Receipts* dapat dilihat *Gross Requirement III*. Dengan demikian pada *Gross Requirement IV* diisi sesuai dengan kebutuhan yang ada di *Gross Requirement III*. Berdasarkan algoritma *Lot-For-Lot* ini maka *Total Cost* yang diperlukan adalah 64.850 Rubel.

D. Lot Sizing Berdasarkan Evolutionary Algorithm dan Solver

Lot Sizing yang dihasilkan dari Algoritma *Lot-For-Lot* memerlukan *Total Cost* sebesar 64.850 Rubel, sedangkan *Lot Sizing* yang dihasilkan dari *Evolutionary Algorithm* dan *Solver* memerlukan *Total Cost* sebesar 62.090 Rubel. Dengan demikian untuk contoh kasus ini *Evolutionary Algorithm* dan *Solver* menghasilkan solusi yang lebih baik dibandingkan dengan Algoritma *Lot-For-Lot*.

4. CONCLUSION

Kombinasi *Evolutionary Algorithm* dan *Solver* dapat digunakan untuk mengoptimasi *Lot Sizing*. Dikarenakan *Evolutionary Algorithm* bekerja menggunakan nilai *random*, *Lot Sizing* yang dihasilkan dari setiap perhitungan dapat berbeda-beda, oleh karena itu perlu dilakukan beberapa kali perhitungan agar dapat diperoleh *Lot Sizing* dengan *Total Cost* paling minimal. Semakin banyak perhitungan yang dilakukan, akan memperbesar peluang didapatkannya *Lot Sizing* dengan *Total Cost* paling minimal, walaupun mengakibatkan waktu penentuan *Lot Sizing* menjadi lebih lama.

Untuk contoh kasus yang digunakan pada penelitian ini, *Evolutionary Algorithm* dan *Solver* menghasilkan *Lot Sizing* yang memerlukan biaya sebesar 62.090 Rubel, sedangkan Algoritma *Lot-For-Lot* menghasilkan *Lot Sizing* yang memerlukan biaya sebesar 64.850 Rubel. Dengan demikian untuk contoh kasus ini *Evolutionary Algorithm* dan *Solver* menghasilkan solusi yang lebih baik dibandingkan dengan Algoritma *Lot-For-Lot*.

Agar kehandalan *Evolutionary Algorithm* dan *Solver* dalam menyelesaikan *Lot Sizing* dapat lebih terukur, maka perlu diujicobakan pada berbagai contoh kasus lainnya dan membandingkannya dengan algoritma-algoritma lainnya seperti *Part-Period Balancing*, *Least Unit Cost*, *Silver Meal*, *Wagner Whitin* dan *Periodic Order Quantity*.

5. REFERENCES

- Agrawal, P., Abutarboush, H. F., Ganesh, T., & Mohamed, A. W. (2021). Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). *IEEE Access*, 9, 26766–26791. <https://doi.org/10.1109/ACCESS.2021.3056407>
- Akbalik, A., Hadj-Alouane, A. B., Sauer, N., & Ghribi, H. (2017). NP-hard and polynomial cases for the single-item lot sizing problem with batch ordering under capacity reservation contract. *European Journal of Operational Research*, 257(2), 483–493. <https://doi.org/10.1016/j.ejor.2016.07.028>
- Badar, A. Q. H. (2021). *Evolutionary Optimization Algorithms*. CRC Press.
- Deng, W., Shang, S., Cai, X., Zhao, H., Song, Y., & Xu, J. (2021). An improved differential evolution algorithm and its application in optimization problem. *Soft Computing*, 25, 5277–5298.
- Dorransoro, B., Yalaoui, F., Talbi, E.-G., & Danoy, G. (2022). *Metaheuristics and Nature Inspired Computing*. Springer.
- Gilbert, J., Koska, O., & Oladi, R. (2023). Excel Sheet for Demonstrating the Specific Factors Model (Solver). *Excel Models for Trade Theory*.
- Jiao, W., Zhang, J. L., & Yan, H. (2017). The stochastic lot-sizing problem with quantity discounts. *Computers and Operations Research*, 80, 1–10. <https://doi.org/10.1016/j.cor.2016.11.014>
- Lu, Q. (2022). *Application of lot-for-lot and order frequency policies for supplier selection and order quantity allocation in two-stage supply chains*.
- Mor, B., Shabtay, D., & Yedidsion, L. (2021). Heuristic algorithms for solving a set of NP-hard single-machine scheduling problems with resource-dependent processing times. *Computers and Industrial Engineering*, 153, 107024. <https://doi.org/10.1016/j.cie.2020.107024>
- Selvi, A. A., Selvabharathi, S. M., & Lavanya, S. (2022). Real Life Optimization Problem using Excel and Solver. *International Journal of Research in Engineering, Science and Management*, 5(5), 155–157.
- Shenoy, D., & Rosas, R. (2018). *Lot-Sizing Heuristics BT - Problems & Solutions in Inventory Management* (D. Shenoy & R. Rosas (eds.); pp. 91–143). Springer International Publishing. https://doi.org/10.1007/978-3-319-65696-0_5