

**STUDI KOMPARATIF ALGORITMA *POPULATION-BASED*
METAHEURISTIC DALAM STUDI KASUS *DNA SEQUENCE ASSEMBLY***

SKRIPSI

Diajukan untuk Memenuhi sebagian dari
Syarat Memperoleh Gelar Sarjana Komputer
Program Studi Ilmu Komputer



Oleh

Yudi Prasetyo

1905348

**PROGRAM STUDI ILMU KOMPUTER
FAKULTAS PENDIDIKAN MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PENDIDIKAN INDONESIA
2023**

**STUDI KOMPARATIF ALGORITMA *POPULATION-BASED*
METAHEURISTIC DALAM STUDI KASUS *DNA SEQUENCE ASSEMBLY***

Oleh

Yudi Prasetyo

NIM 1905348

Sebuah Skripsi yang Diajukan untuk Memenuhi Salah Satu Syarat Memperoleh
Gelar Sarjana Komputer di Fakultas Pendidikan Matematika dan Ilmu
Pengetahuan Alam

© Yudi Prasetyo 2023

Universitas Pendidikan Indonesia

Agustus 2023

Hak cipta dilindungi undang-undang
Skripsi ini tidak boleh diperbanyak seluruhnya atau sebagian, dengan dicetak
ulang, difoto kopi, atau cara lainnya tanpa izin dari penulis

Yudi Prasetyo, 2023

*STUDI KOMPARATIF ALGORITMA POPULATION-BASED METAHEURISTIC DALAM STUDI KASUS DNA
SEQUENCE ASSEMBLY*

Universitas Pendidikan Indonesia | repository.upi.edu | perpustakaan.upi.edu

YUDI PRASETYO

1905348

**STUDI KOMPARATIF ALGORITMA *POPULATION-BASED METAHEURISTIC*
DALAM STUDI KASUS *DNA SEQUENCE ASSEMBLY***

DISETUJUI DAN DISAHKAN OLEH PEMBIMBING:
Pembimbing I,



Prof. Dr. Lala Septem Riza, M.T.
NIP. 197809262008121001

Pembimbing II,



Herbert Siregar, M.T.
NIP. 197005022008121001

Mengetahui,
Ketua Program Studi Ilmu Komputer



Dr. Muhamad Nursalman, M.T.
NIP. 197909292006041002

STUDI KOMPARATIF ALGORITMA *POPULATION-BASED METAHEURISTIC* DALAM STUDI KASUS *DNA SEQUENCE ASSEMBLY*

Oleh

Yudi Prasetyo

NIM 1905348

ABSTRAK

Teknologi modern menemui kesulitan dalam melakukan DNA sequencing pada sekuens DNA yang panjang. Karena itu, sekuens DNA yang lebih panjang harus dipotong menjadi fragmen-fragmen lebih kecil. *DNA sequence assembly* adalah proses penggabungan sejumlah sekuens *genome* yang pendek untuk menciptakan urutan DNA yang lebih panjang. Tujuan dari studi ini adalah membandingkan kinerja dari beberapa algoritma *population-based metaheuristic* dalam menangani masalah *DNA sequence assembly* berdasarkan waktu komputasi, jumlah contigs, dan nilai *overlap*. Algoritma yang digunakan dalam studi ini meliputi *Honey Badger Algorithm* (HBA), *Lévy Flight Distribution* (LFD), *African Vultures Optimization Algorithm* (AVOA) dan *Particle Swarm Optimization* (PSO). Secara keseluruhan, AVOA memiliki hasil yang terbaik di mana AVOA dapat menghasilkan total *overlap* yang paling banyak, di mana *overlap* terbanyak adalah 49952 di dataset dengan *length* 750 dan *coverage* 25. AVOA juga memiliki efisiensi yang terbaik karena memiliki waktu komputasi yang lebih cepat dibanding algoritma lain di semua dataset yang digunakan. Selain AVOA, PSO juga menghasilkan total *overlap* dan waktu komputasi yang tidak jauh dari AVOA. Namun jika berdasarkan jumlah *contig*, HBA mampu menghasilkan jumlah *contig* yang paling sedikit terutama pada dataset dengan *length* 750 dan *coverage* 15 dengan jumlah 6 contig.

Kata kunci: bahasa pemrograman R, *DNA sequence assembly*, optimasi, *population-based metaheuristic*, *string matching*

COMPARATIVE STUDY OF POPULATION-BASED METAHEURISTIC ALGORITHMS IN CASE STUDY OF DNA SEQUENCE ASSEMBLY

Arranged by

Yudi Prasetyo

NIM 1905348

ABSTRACT

Modern technology encounters difficulties in performing DNA sequencing on long DNA sequences. Therefore, longer DNA sequences must be cut into smaller fragments. DNA sequence assembly is the process of combining a number of short genome sequences to create a longer DNA sequence. The purpose of this study is to compare the performance of several population-based metaheuristic algorithms in handling the DNA sequence assembly problem based on computation time, number of contigs, and overlap value. The algorithms used in this study include Honey Badger Algorithm (HBA), Lévy Flight Distribution (LFD), African Vultures Optimization Algorithm (AVOA) and Particle Swarm Optimization (PSO). Overall, AVOA has the best results where it can produce the most total overlap, where the most overlap is 49952 in the dataset with length 750 and coverage 25. AVOA also has the best efficiency because it has a faster computation time than other algorithms in all datasets used. Besides AVOA, PSO also produces total overlap and computation time that is not far from AVOA. However, if based on the number of contigs, HBA is able to produce the least number of contigs, especially on datasets with length 750 and coverage 15 with a total of 6 contigs.

Keywords: DNA sequence assembly, optimization, population-based metaheuristic, R programming language, string matching

DAFTAR ISI

ABSTRAK	i
ABSTRACT	ii
KATA PENGANTAR	iii
UCAPAN TERIMAKASIH	iv
DAFTAR ISI	vi
DAFTAR TABEL	ix
DAFTAR GAMBAR	x
DAFTAR LAMPIRAN	i
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	6
1.3 Tujuan Penelitian	6
1.4 Manfaat Penelitian	6
1.5 Batasan Masalah	7
1.6 Sistematika Penulisan	7
BAB II KAJIAN PUSTAKA	9
2.1 DNA dan DNA Assembly	9
2.1.1 DNA <i>Sequencing</i>	10
2.1.2 Definisi DNA Assembly.....	13
2.1.3 Metode DNA Assembly	13
2.2 Algoritma String Matching.....	14
2.3 Bahasa Pemrograman R.....	16
2.3.1 Sejarah R	18
2.3.2 RStudio	20
2.4 Optimasi	22
2.4.1 Pemanfaatan Optimasi	23

2.4.2	Klasifikasi Permasalahan Optimasi	25
2.4.3	Metode Optimasi	27
2.5	Metaheuristics	30
2.5.1	Population-based Metaheuristic	31
2.5.2	Honey Badger Algorithm	33
2.5.3	Particle Swarm Optimization	37
2.5.4	African Vultures Optimization Algorithm	40
2.5.5	Lévy Flight Distribution	42
BAB III METODE PENELITIAN		48
3.1	Desain Penelitian	48
3.2	Metode Penelitian	50
3.2.1	Metode Pengumpulan Data	50
3.2.2	Metode Pengembangan Perangkat Lunak	51
3.3	Alat dan Bahan Penelitian	52
3.3.1	Alat Penelitian	52
3.3.2	Bahan Penelitian	53
BAB IV HASIL DAN PEMBAHASAN.....		54
4.1	Pengumpulan Data.....	54
4.1.1	Mengunduh Data dari NCBI	54
4.1.2	Penjelasan Isi File	55
4.2	Perancangan Model	55
4.2.1	<i>Input DNA reads</i> dan Masukkan Pengguna	57
4.2.2	Inisialisasi <i>Buffer Table</i>	59
4.2.3	Population-based Metaheuristic	61
4.2.4	<i>Assemble DNA</i>	67
4.2.5	<i>Reassemble DNA</i>	69

4.3	Implementasi Model Komputasi	70
4.3.1	Input DNA <i>reads</i>	70
4.3.2	Inisialisasi <i>Buffer Table</i>	71
4.3.3	<i>Population-based metaheuristic</i>	71
4.3.4	<i>Assemble DNA</i>	75
4.3.5	<i>Reassemble DNA</i>	76
4.4	Rancangan Skenario Eksperimen	77
4.4.1	Eksperimen menggunakan data <i>dummy</i>	77
4.4.2	Eksperimen menggunakan data <i>benchmark</i>	78
4.4.3	Instrumen Evaluasi	79
4.5	Hasil Eksperimen.....	79
4.5.1	Hasil eksperimen menggunakan data <i>dummy</i>	79
4.5.2	Hasil eksperimen menggunakan data <i>benchmark</i>	82
4.6	Pembahasan	88
BAB V HASIL DAN PEMBAHASAN		93
5.1	Kesimpulan.....	93
5.2	Saran	94
DAFTAR PUSTAKA		95
LAMPIRAN.....		108

DAFTAR PUSTAKA

- Abdel-Basset, M., Abdel-Fatah, L., & Sangaiyah, A. K. (2018). Metaheuristic algorithms: A comprehensive review. In *Computational Intelligence for Multimedia Big Data on the Cloud with Engineering Applications* (pp. 185–231). Elsevier. <https://doi.org/10.1016/B978-0-12-813314-9.00010-4>
- Abdollahzadeh, B., Gharehchopogh, F. S., & Mirjalili, S. (2021). African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems. *Computers & Industrial Engineering*, *158*, 107408. <https://doi.org/10.1016/j.cie.2021.107408>
- Ahmadianfar, I., Adib, A., & Salarijazi, M. (2016). Optimizing multireservoir operation: Hybrid of bat algorithm and differential evolution. *Journal of Water Resources Planning and Management*, *142*(2), 05015010. [https://doi.org/10.1061/\(ASCE\)WR.1943-5452.0000606](https://doi.org/10.1061/(ASCE)WR.1943-5452.0000606)
- Akopyan, A. V. (2015). Geometry of the cardioid. *The American Mathematical Monthly*, *122*(2), 144–150.
- Alba, E., & Luque, G. (2007). A new local search algorithm for the DNA fragment assembly problem. In C. Cotta & J. van Hemert (Eds.), *Evolutionary Computation in Combinatorial Optimization* (Vol. 4446, pp. 1–12). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-71615-0_1
- Alba, E., & Luque, G. (2008). A hybrid genetic algorithm for the DNA fragment assembly problem. In C. Cotta & J. Van Hemert (Eds.), *Recent Advances in Evolutionary Computation for Combinatorial Optimization* (Vol. 153, pp. 101–112). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-70807-0_7
- Alkan, C., Sajjadian, S., & Eichler, E. E. (2011). Limitations of next-generation genome sequence assembly. *Nature Methods*, *8*(1), 61–65. <https://doi.org/10.1038/nmeth.1527>
- Allaire, J., Xie, Y., McPherson, J., Luraschi, J., Ushey, K., Atkins, A., & Iannone, R. (2018). *Rmarkdown: Dynamic documents for R*. R Package Version 1.15.

- Bankevich, A., Nurk, S., Antipov, D., Gurevich, A. A., Dvorkin, M., Kulikov, A. S., Lesin, V. M., Nikolenko, S. I., Pham, S., Pribelski, A. D., Pyshkin, A. V., Sirotkin, A. V., Vyahhi, N., Tesler, G., Alekseyev, M. A., & Pevzner, P. A. (2012). Spades: A new genome assembly algorithm and its applications to single-cell sequencing. *Journal of Computational Biology*, *19*(5), 455–477. <https://doi.org/10.1089/cmb.2012.0021>
- Baumer, B., Cetinkaya-Rundel, M., Bray, A., Loi, L., & Horton, N. J. (2014). *R markdown: Integrating a reproducible analysis tool into introductory statistics*. <https://doi.org/10.48550/ARXIV.1402.1894>
- Becker, R. A., & Chambers, J. M. (1984). *S: An interactive environment for data analysis and graphics*. Wadsworth Advanced Book Program.
- Begg, C. M., Begg, K. S., Du Toit, J. T., & Mills, M. G. L. (2003). Scent-marking behaviour of the honey badger, *Mellivora capensis* (Mustelidae), in the southern Kalahari. *Animal Behaviour*, *66*(5), 917–929. <https://doi.org/10.1006/anbe.2003.2223>
- Begg, C. M., Begg, K. S., Toit, J. T., & Mills, M. G. L. (2005). Life-history variables of an atypical mustelid, the honey badger *Mellivora capensis*. *Journal of Zoology*, *265*(1), 17–22. <https://doi.org/10.1017/S0952836904005990>
- Beheshti, Z., & Shamsuddin, S. M. H. (2013). A review of population-based meta-heuristic algorithms. *Int. j. Adv. Soft Comput. Appl*, *5*(1), 1–35.
- Ben Ali, A., Luque, G., & Alba, E. (2020). An efficient discrete PSO coupled with a fast local search heuristic for the DNA fragment assembly problem. *Information Sciences*, *512*, 880–908. <https://doi.org/10.1016/j.ins.2019.10.026>
- Bhagya Sri, M., Bhavsar, R., & Narooka, P. (2018). String matching algorithms. *International Journal Of Engineering And Computer Science*, *7*(03), Article 03. <https://doi.org/10.18535/ijecs/v7i3.19>
- Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys*, *35*(3), 268–308. <https://doi.org/10.1145/937503.937505>

- Boyd, D., & Crawford, K. (2012). Critical questions for big data: Provocations for a cultural, technological, and scholarly phenomenon. *Information, Communication & Society*, 15(5), Article 5.
- Boyd, S. P., & Vandenberghe, L. (2004). *Convex optimization*. Cambridge University Press.
- Bucur, D. (2017). A stochastic de novo assembly algorithm for viral-sized genomes obtains correct genomes and builds consensus. *Information Sciences*, 420, 184–199. <https://doi.org/10.1016/j.ins.2017.07.039>
- Cauchy, A. (1847). Methode generale pour la resolution des systemes d'equations simultanees. *C.R. Acad. Sci. Paris*, 25, 536–538.
- Cavalli-Sforza, L. L., & Feldman, M. W. (2003). The application of molecular genetic approaches to the study of human evolution. *Nature Genetics*, 33(S3), 266–275. <https://doi.org/10.1038/ng1113>
- Chambers, J. M. (1998). *Programming with data: A guide to the S language*. Springer.
- Charras, C., & Lecroq, T. (2004). *Handbook of exact string matching algorithms*. King's College.
- Clancy, S. (2008). DNA damage & repair: Mechanisms for maintaining DNA integrity. *Nature Education*, 1(1), 103.
- Clerc, M. (2006). *Particle swarm optimization*. ISTE.
- Conn, A. R., Scheinberg, K., & Toint, Ph. L. (1997). Recent progress in unconstrained nonlinear optimization without derivatives. *Mathematical Programming*, 79(1–3), 397–414. <https://doi.org/10.1007/BF02614326>
- Crick, F. (1970). Central dogma of molecular biology. *Nature*, 227(5258), 561–563. <https://doi.org/10.1038/227561a0>
- Deb, K. (2011). Multi-objective optimisation using evolutionary algorithms: An introduction. In L. Wang, A. H. C. Ng, & K. Deb (Eds.), *Multi-objective Evolutionary Optimisation for Product Design and Manufacturing* (pp. 3–34). Springer London. https://doi.org/10.1007/978-0-85729-652-8_1
- Denisov, G., Walenz, B., Halpern, A. L., Miller, J., Axelrod, N., Levy, S., & Sutton, G. (2008). Consensus generation and variant detection by celera

- assembler. *Bioinformatics*, 24(8), 1035–1040.
<https://doi.org/10.1093/bioinformatics/btn074>
- Dohm, J. C., Lottaz, C., Borodina, T., & Himmelbauer, H. (2007). Sharcgs: A fast and highly accurate short-read assembly algorithm for de novo genomic sequencing. *Genome Research*, 17(11), 1697–1706.
<https://doi.org/10.1101/gr.6435207>
- Dorigo, M., Maniezzo, V., & Colormi, A. (1996). Ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 26(1), 29–41.
<https://doi.org/10.1109/3477.484436>
- Dorigo, M., & Stützle, T. (2004). *Ant colony optimization*. MIT Press.
- Dorransoro, B., Alba, E., Luque, G., & Bouvry, P. (2008). A self-adaptive cellular memetic algorithm for the DNA fragment assembly problem. *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, 2651–2658.
<https://doi.org/10.1109/CEC.2008.4631154>
- Dorransoro, B., Bouvry, P., & Alba, E. (2010). Iterated local search for de novo genomic sequencing. In L. Rutkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh, & J. M. Zurada (Eds.), *Artificial Intelligence and Soft Computing* (Vol. 6114, pp. 428–436). Springer Berlin Heidelberg.
https://doi.org/10.1007/978-3-642-13232-2_52
- Duchi, J., Hazan, E., & Singer, Y. (2011). Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(7).
- Eiben, A. E., & Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer Berlin Heidelberg. <https://doi.org/10.1007/978-3-662-44874-8>
- Fox, J., Weisberg, S., & Fox, J. (2011). *An R companion to applied regression* (2nd ed). SAGE Publications.
- Gandomi, A. H., Yang, X.-S., Talatahari, S., & Alavi, A. H. (Eds.). (2013). *Metaheuristic applications in structures and infrastructures* (First edition). Elsevier.

- Gheraibia, Y., Moussaoui, A., Kabir, S., & Mazouzi, S. (2016). Pe-DFA: Penguins search optimisation algorithm for dna fragment assembly. *International Journal of Applied Metaheuristic Computing*, 7(2), 58–70. <https://doi.org/10.4018/IJAMC.2016040104>
- Glover, F. (1986a). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5), 533–549. [https://doi.org/10.1016/0305-0548\(86\)90048-1](https://doi.org/10.1016/0305-0548(86)90048-1)
- Glover, F. (1986b). Future paths for integer programming and links to artificial intelligence. *Computers & Operations Research*, 13(5), 533–549. [https://doi.org/10.1016/0305-0548\(86\)90048-1](https://doi.org/10.1016/0305-0548(86)90048-1)
- Goldberg, D. E. (1989). Genetic algorithms in search. *Optimization, Machine Learning*.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. The MIT Press.
- Hashim, F. A., Houssein, E. H., Hussain, K., Mabrouk, M. S., & Al-Atabany, W. (2022). Honey badger algorithm: New metaheuristic algorithm for solving optimization problems. *Mathematics and Computers in Simulation*, 192, 84–110. <https://doi.org/10.1016/j.matcom.2021.08.013>
- Hegazy, T. (1999). Optimization of resource allocation and leveling using genetic algorithms. *Journal of Construction Engineering and Management*, 125(3), 167–175. [https://doi.org/10.1061/\(ASCE\)0733-9364\(1999\)125:3\(167\)](https://doi.org/10.1061/(ASCE)0733-9364(1999)125:3(167))
- Heptner, V. G., & Robert, S. (2001). *Mammals of the soviet union: Vol. II., part Ib carnivora (weasels; additional species)*. Smithsonian Institution Libraries and the National Science Foundation.
- Holland, J. H. (1992). *Adaptation in natural and artificial systems: An introductory analysis with applications to biology, control, and artificial intelligence* (1st MIT Press ed). MIT Press.
- Houssein, E. H., Saad, M. R., Hashim, F. A., Shaban, H., & Hassaballah, M. (2020). Lévy flight distribution: A new metaheuristic algorithm for solving engineering optimization problems. *Engineering Applications of Artificial Intelligence*, 94, 103731. <https://doi.org/10.1016/j.engappai.2020.103731>

- Huang, K.-W., Chen, J.-L., & Yang, C.-S. (2012). A hybrid PSO-based algorithm for solving DNA fragment assembly problem. *2012 Third International Conference on Innovations in Bio-Inspired Computing and Applications*, 223–228. <https://doi.org/10.1109/IBICA.2012.8>
- Huang, K.-W., Chen, J.-L., Yang, C.-S., & Tsai, C.-W. (2015). A memetic particle swarm optimization algorithm for solving the DNA fragment assembly problem. *Neural Computing and Applications*, 26(3), 495–506. <https://doi.org/10.1007/s00521-014-1659-0>
- Huang, X., & Madan, A. (1999). CAP3: A DNA sequence assembly program. *Genome Research*, 9(9), 868–877. <https://doi.org/10.1101/gr.9.9.868>
- Hughes, J. A., Houghten, S., & Ashlock, D. (2016). Restarting and recentering genetic algorithm variations for DNA fragment assembly: The necessity of a multi-strategy approach. *Biosystems*, 150, 35–45. <https://doi.org/10.1016/j.biosystems.2016.08.001>
- Ihaka, R., & Gentleman, R. (1996). R: A language for data analysis and graphics. *Journal of Computational and Graphical Statistics*, 5(3), 299–314. <https://doi.org/10.1080/10618600.1996.10474713>
- Indumathy, R., Uma Maheswari, S., & Subashini, G. (2015). Nature-inspired novel cuckoo search algorithm for genome sequence assembly. *Sadhana*, 40(1), 1–14. <https://doi.org/10.1007/s12046-014-0300-3>
- Kapner, D. J., Cook, T. S., Adelberger, E. G., Gundlach, J. H., Heckel, B. R., Hoyle, C. D., & Swanson, H. E. (2007). Tests of the gravitational inverse-square law below the dark-energy length scale. *Physical Review Letters*, 98(2), 021101. <https://doi.org/10.1103/PhysRevLett.98.021101>
- Karaboga, D., & Akay, B. (2009). A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation*, 214(1), 108–132. <https://doi.org/10.1016/j.amc.2009.03.090>
- Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *Proceedings of ICNN'95 - International Conference on Neural Networks*, 4, 1942–1948. <https://doi.org/10.1109/ICNN.1995.488968>
- Kingma, D. P., & Ba, J. (2014). *Adam: A method for stochastic optimization*. <https://doi.org/10.48550/ARXIV.1412.6980>

- Kirkpatrick, S., Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science*, 220(4598), 671–680.
<https://doi.org/10.1126/science.220.4598.671>
- Knuth, D. E., Morris, Jr., J. H., & Pratt, V. R. (1977). Fast pattern matching in strings. *SIAM Journal on Computing*, 6(2), Article 2.
<https://doi.org/10.1137/0206024>
- Kochenderfer, M. J., & Wheeler, T. A. (2019). *Algorithms for optimization*. The MIT Press.
- Koren, S., Walenz, B. P., Berlin, K., Miller, J. R., Bergman, N. H., & Phillippy, A. M. (2017). Canu: Scalable and accurate long-read assembly via adaptive k -mer weighting and repeat separation. *Genome Research*, 27(5), 722–736. <https://doi.org/10.1101/gr.215087.116>
- Kunin, V., Copeland, A., Lapidus, A., Mavromatis, K., & Hugenholtz, P. (2008). A bioinformatician's guide to metagenomics. *Microbiology and Molecular Biology Reviews*, 72(4), 557–578. <https://doi.org/10.1128/MMBR.00009-08>
- Lai, B., Ding, R., Li, Y., Duan, L., & Zhu, H. (2012). A de novo metagenomic assembly program for shotgun DNA reads. *Bioinformatics*, 28(11), 1455–1462. <https://doi.org/10.1093/bioinformatics/bts162>
- Li, R., Zhu, H., Ruan, J., Qian, W., Fang, X., Shi, Z., Li, Y., Li, S., Shan, G., Kristiansen, K., Li, S., Yang, H., Wang, J., & Wang, J. (2010). De novo assembly of human genomes with massively parallel short read sequencing. *Genome Research*, 20(2), 265–272.
<https://doi.org/10.1101/gr.097261.109>
- Lindahl, T. (1993). Instability and decay of the primary structure of DNA. *Nature*, 362(6422), 709–715. <https://doi.org/10.1038/362709a0>
- Liu, X., Pande, P. R., Meyerhenke, H., & Bader, D. A. (2013). Pasqual: Parallel techniques for next generation genome sequence assembly. *IEEE Transactions on Parallel and Distributed Systems*, 24(5), 977–986.
<https://doi.org/10.1109/TPDS.2012.190>

- Magdziarz, M., & Szczotka, W. (2016). Quenched trap model for lévy flights. *Communications in Nonlinear Science and Numerical Simulation*, 30(1–3), 5–14. <https://doi.org/10.1016/j.cnsns.2015.05.027>
- Mallen-Fullerton, G. M., & Fernandez-Anaya, G. (2013). DNA fragment assembly using optimization. *2013 IEEE Congress on Evolutionary Computation*, 1570–1577. <https://doi.org/10.1109/CEC.2013.6557749>
- Mardis, E. R. (2008). Next-generation DNA sequencing methods. *Annual Review of Genomics and Human Genetics*, 9(1), 387–402. <https://doi.org/10.1146/annurev.genom.9.081307.164359>
- Mathews, G. B. (1896). On the partition of numbers. *Proceedings of the London Mathematical Society*, s1-28(1), 486–490. <https://doi.org/10.1112/plms/s1-28.1.486>
- McNally, S., Roche, J., & Caton, S. (2018). Predicting the price of bitcoin using machine learning. *2018 26th Euromicro International Conference on Parallel, Distributed and Network-Based Processing (PDP)*, 339–343. <https://doi.org/10.1109/PDP2018.2018.00060>
- Meksangsouy, P., & Chaiyaratana, N. (2003). DNA fragment assembly using an ant colony system algorithm. *The 2003 Congress on Evolutionary Computation, 2003. CEC '03.*, 3, 1756–1763. <https://doi.org/10.1109/CEC.2003.1299885>
- Meselson, M., & Stahl, F. W. (1958). The replication of DNA in *Escherichia coli*. *Proceedings of the National Academy of Sciences*, 44(7), 671–682. <https://doi.org/10.1073/pnas.44.7.671>
- Miller, J. R., Koren, S., & Sutton, G. (2010). Assembly algorithms for next-generation sequencing data. *Genomics*, 95(6), 315–327. <https://doi.org/10.1016/j.ygeno.2010.03.001>
- Minetti, G., & Alba, E. (2010). Metaheuristic assemblers of DNA strands: Noiseless and noisy cases. *IEEE Congress on Evolutionary Computation*, 1–8. <https://doi.org/10.1109/CEC.2010.5586524>
- Minetti, G., Leguizamón, G., & Alba, E. (2014). An improved trajectory-based hybrid metaheuristic applied to the noisy DNA fragment assembly

- problem. *Information Sciences*, 277, 273–283.
<https://doi.org/10.1016/j.ins.2014.02.020>
- Nagarajan, N., & Pop, M. (2013). Sequence assembly demystified. *Nature Reviews Genetics*, 14(3), 157–167. <https://doi.org/10.1038/nrg3367>
- Nebro, A. J., Luque, G., Luna, F., & Alba, E. (2008). DNA fragment assembly using a grid-based genetic algorithm. *Computers & Operations Research*, 35(9), 2776–2790. <https://doi.org/10.1016/j.cor.2006.12.011>
- Papageorgiou, M., Kiakaki, C., Dinopoulou, V., Kotsialos, A., & Yibing Wang. (2003). Review of road traffic control strategies. *Proceedings of the IEEE*, 91(12), 2043–2067. <https://doi.org/10.1109/JPROC.2003.819610>
- Pearl, J. (1984). *Heuristics: Intelligent search strategies for computer problem solving*. Addison-Wesley Longman Publishing Co., Inc.
- Peng, R. D. (2014). *R programming for data science*. Leanpub.
- Pevzner, P. A., Tang, H., & Waterman, M. S. (2001). An eulerian path approach to DNA fragment assembly. *Proceedings of the National Academy of Sciences*, 98(17), 9748–9753. <https://doi.org/10.1073/pnas.171285098>
- Piryonesi, S. M., & Tavakolan, M. (2017). A mathematical programming model for solving cost-safety optimization (CSO) problems in the maintenance of structures. *KSCE Journal of Civil Engineering*, 21(6), 2226–2234. <https://doi.org/10.1007/s12205-017-0531-z>
- Pressman, R. S. (2010). A practitioner’s approach. *Software Engineering*, 2, 41–42.
- R Development Core Team. (2010). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*.
- Racine, J. S. (2012). Rstudio: A platform-independent IDE for R and sweave: Software review. *Journal of Applied Econometrics*, 27(1), 167–172. <https://doi.org/10.1002/jae.1278>
- Rajagopal, I., & Maheswari Sankareswaran, U. (2015). An adaptive particle swarm optimization algorithm for solving DNA fragment assembly problem. *Current Bioinformatics*, 10(1), 97–105.
- Rao, S. S. (2020). *Engineering optimization: Theory and practice* (Fifth edition). Wiley.

- Reynolds, C. W. (1987). *Flocks, herds and schools: A distributed behavioral model*. 25–34.
- Riza, L. S., Nugroho, E. P., & others. (2018). Metaheuristicopt: An R package for optimisation based on meta-heuristics algorithms. *Pertanika Journal of Science & Technology*, 26(3).
- RStudio Team. (2020). *Rstudio: Integrated development environment for R*. RStudio, PBC. <http://www.rstudio.com/>
- Sanger, F., Nicklen, S., & Coulson, A. R. (1977). DNA sequencing with chain-terminating inhibitors. *Proceedings of the National Academy of Sciences*, 74(12), 5463–5467. <https://doi.org/10.1073/pnas.74.12.5463>
- Sharma, G. S., Singh, M., & Singh, T. (2011). Optimization of energy in robotic arm using genetic algorithm. *International Journal of Computer Science and Technology*, 2(2), 315–317.
- Shi, Y., & Eberhart, R. (1998). *A modified particle swarm optimizer*. 69–73.
- Simpson, J. T., Wong, K., Jackman, S. D., Schein, J. E., Jones, S. J. M., & Birol, I. (2009). ABySS: A parallel assembler for short read sequence data. *Genome Research*, 19(6), 1117–1123. <https://doi.org/10.1101/gr.089532.108>
- Sörensen, K., & Glover, F. (2013). Metaheuristics. *Encyclopedia of Operations Research and Management Science*, 62, 960–970.
- Storn, R., & Price, K. (1997). Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, 11, 341–359.
- Sutskever, I., Martens, J., Dahl, G., & Hinton, G. (2013). On the importance of initialization and momentum in deep learning. In S. Dasgupta & D. McAllester (Eds.), *Proceedings of the 30th International Conference on Machine Learning* (Vol. 28, Issue 3, pp. 1139–1147). PMLR. <https://proceedings.mlr.press/v28/sutskever13.html>
- Sutton, G. G., White, O., Adams, M. D., & Kerlavage, A. R. (1995). Tigr assembler: A new tool for assembling large shotgun sequencing projects. *Genome Science and Technology*, 1(1), 9–19. <https://doi.org/10.1089/gst.1995.1.9>

- Talbi, E.-G. (2009). *Metaheuristics: From design to implementation*. John Wiley & Sons.
- Tambunan, A. P. (2008). *Menilai harga wajar saham*. Elex Media Komputindo.
- Tieleman, T., & Hinton, G. (2012). Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural Networks for Machine Learning*, 4(2), 26–31.
- Tsai, L.-W. (1999). *Robot analysis: The mechanics of serial and parallel manipulators*. John Wiley & Sons.
- Verma, R. S., & Kumar, S. (2012). DSAPSO: DNA sequence assembly using continuous particle swarm optimization with smallest position value rule. *2012 1st International Conference on Recent Advances in Information Technology (RAIT)*, 410–415. <https://doi.org/10.1109/RAIT.2012.6194455>
- Vissers, J., & Beech, R. (2005). *Health operations management: Patient flow logistics in health care*. Routledge.
- Voelkerding, K. V., Dames, S. A., & Durtschi, J. D. (2009). Next-generation sequencing: From basic research to diagnostics. *Clinical Chemistry*, 55(4), 641–658. <https://doi.org/10.1373/clinchem.2008.112789>
- Vukcevic, D. (2018). Extending a constrained hybrid dynamics solver for energy-optimal robot motions in the presence of static friction. In *Technical Report / Hochschule Bonn-Rhein-Sieg—University of Applied Sciences: Vol. Department of Computer Science* (p. 2239 KB, 66 pages) [Application/pdf]. Hochschule Bonn-Rhein-Sieg. <https://doi.org/10.18418/978-3-96043-063-6>
- Wang, D., Tan, D., & Liu, L. (2018). Particle swarm optimization algorithm: An overview. *Soft Computing*, 22(2), 387–408. <https://doi.org/10.1007/s00500-016-2474-6>
- Warren, R. L., Sutton, G. G., Jones, S. J. M., & Holt, R. A. (2007). Assembling millions of short DNA sequences using SSAKE. *Bioinformatics*, 23(4), 500–501. <https://doi.org/10.1093/bioinformatics/btl629>
- Watson, J. D., & Crick, F. H. C. (1953). Molecular structure of nucleic acids: A structure for deoxyribose nucleic acid. *Nature*, 171(4356), 737–738. <https://doi.org/10.1038/171737a0>

- Wickham, H. (2009). Elegant graphics for data analysis (ggplot2). *Applied Spatial Data Analysis R*, 784, 785.
- Wickham, H. (2010). A layered grammar of graphics. *Journal of Computational and Graphical Statistics*, 19(1), 3–28.
<https://doi.org/10.1198/jcgs.2009.07098>
- Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L., François, R., Grolemund, G., Hayes, A., Henry, L., Hester, J., Kuhn, M., Pedersen, T., Miller, E., Bache, S., Müller, K., Ooms, J., Robinson, D., Seidel, D., Spinu, V., ... Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, 4(43), 1686. <https://doi.org/10.21105/joss.01686>
- Wickham, H., François, R., Henry, L., Müller, K., & Wickham, M. H. (2019). Package ‘dplyr.’ *A Grammar of Data Manipulation. R Package Version*, 8.
- Williamson, D. P., & Shmoys, D. B. (2011). *The design of approximation algorithms*. Cambridge University Press.
- Wilson, J. H., & Hunt, T. (2015). *Molecular biology of the cell: The problems book* (Sixth edition). Garland Science/Taylor & Francis Group.
- Xie, Y. (2013). *Dynamic documents with R and knitr* chapman & hall/crc the R series. Taylor & Francis, Boca Raton, Florida.
- Xie, Y., Allaire, J. J., & Grolemund, G. (2018). *R markdown: The definitive guide* (1st ed.). Chapman and Hall/CRC. <https://doi.org/10.1201/9781138359444>
- Yang, X.-S. (Ed.). (2013). *Swarm intelligence and bio-inspired computation: Theory and applications* (1st ed). Elsevier.
- Yang, X.-S. & Suash Deb. (2009). Cuckoo search via lévy flights. *2009 World Congress on Nature & Biologically Inspired Computing (NaBIC)*, 210–214. <https://doi.org/10.1109/NABIC.2009.5393690>
- Zeiler, M. D. (2012). *Adadelta: An adaptive learning rate method*.
<https://doi.org/10.48550/ARXIV.1212.5701>
- Zemali, E., & Boukra, A. (2018). CS-ABC: A cooperative system based on artificial bee colony to resolve the DNA fragment assembly problem. *International Journal of Data Mining and Bioinformatics*, 21(2), 145.
<https://doi.org/10.1504/IJDMB.2018.096407>

- Zerbino, D. R., & Birney, E. (2008). Velvet: Algorithms for de novo short read assembly using de bruijn graphs. *Genome Research*, 18(5), 821–829. <https://doi.org/10.1101/gr.074492.107>
- Zhang, J.-R., Zhang, J., Lok, T.-M., & Lyu, M. R. (2007). A hybrid particle swarm optimization–back-propagation algorithm for feedforward neural network training. *Applied Mathematics and Computation*, 185(2), 1026–1037. <https://doi.org/10.1016/j.amc.2006.07.025>
- Zhang, Z., Schwartz, S., Wagner, L., & Miller, W. (2000). A greedy algorithm for aligning DNA sequences. *Journal of Computational Biology*, 7(1–2), 203–214. <https://doi.org/10.1089/10665270050081478>