LITERATURE REVIEW

PENELITIAN MANDIRI DALAM SAINS KOMPUTASI III - IV

Mohammad Rizka Fadhli 20921004

Sains Komputasi ITB

LITERATURE REVIEW

Paper I

Metode Eksak

Pada penelitian mandiri I, pada *literature review* [1] saya mendapatkan beberapa informasi sebagai berikut:

- Metode *simplex* adalah metode eksak yang digunakan untuk menyelesaikan *linear programming*. Solusi yang dihasilkan merupakan bilangan *real* atau kontinu.
- Pada *MILP*, variabel yang terlibat sangat beragam (*integer*, *binary*, dan kontinu). Membulatkan bilangan solusi *linear programming* untuk mendapatkan solusi *integer* atau *binary* dari suatu masalah *MILP* tidak menjamin keoptimalan tercapai.

Oleh karena itu, perlu pendekatan tertentu dari *linear programming* agar hasilnya bisa digunakan di *MILP*. Salah satu pendekatan yang bisa dilakukan adalah melakukan *constraint relaxation* [2].

Metode *Meta Heuristic*: SOA untuk MINLP

Solving Mixed Integer Nonlinear Programming Problems Using Spiral Dynamics Optimization Algorithm

Adhe Kania[1, a)] and Kuntjoro Adji Sidarto[1,b)]

[1] Department of Mathematics, Institut Teknologi Bandung, Bandung 40132, Indonesia.

[a)] Corresponding author: adhe.kania@math.itb.ac.id
[b)] sidarto@math.itb.ac.id

Abstract. Many engineering and practical problem can be modeled by mixed integer nonlinear programming. This paper proposes to solve the problem with modified spiral dynamics inspired optimization method of Tamura and Yasuda. Four test cases have been examined, including problem in engineering and sport. This method succeeds in obtaining the optimal result in all test cases.

[3]

Key Take Points I

Salah satu trik yang bisa dilakukan agar SOA bisa menyelesaikan *mixed integer programming* adalah dengan mengubah *constrained optimization problem* menjadi *unconstrained optimization problem* kemudian memanfaatkan *penalty constant*.

Misal suatu permasalahan MILP atau MINLP bisa ditulis secara umum sebagai berikut:

$$\min_{x \in \mathbb{R}^n} f(x)$$

$$\text{subject to: } g_i(x) = 0, i = 1, 2, .., M$$

$$\text{and } h_j(x) \leq 0, i = 1, 2, .., N$$

Key Take Points II

$$x = (x_1, x_2, ..., x_n)^T \in \mathbb{N}$$

Bentuk di atas bisa kita ubah menjadi:

$$F(x, \alpha, \beta) = f(x) + \sum_{i=1}^{M} \alpha_i g_i^2(x) + \sum_{j=1}^{N} \beta_j (\max(h_i(x), 0))^2$$

dimana α, β merupakan *penalty constant* yang bisa dibuat sangat besar.

Permasalahan yang Ditemui pada Paper Ini

SOA relatif mudah untuk dituliskan dalam bentuk algoritma bahasa pemrograman manapun. Tapi ada satu hal yang bisa menjadi batu ganjalan dalam menuliskan algoritmanya. Apa itu? Yaitu pendefinisian matriks rotasi untuk masalah dengan n-dimensi.

Bentuk umum dari matriks rotasi adalah sebagai berikut:

$$R^{(n)}(\theta_{1,2}, \theta_{1,3}, .., \theta_{n,n-1}) = \prod_{i=1}^{n-1} \left(\prod_{j=1}^{i} R^{(n)}_{n-i,n+1-j}(\theta_{n-i,n+1-j}) \right)$$

Apakah ini *cross product* atau *dot product*?

Paper II

SOA

Spiral Dynamics Inspired Optimization

Kenichi Tamura and Keiichiro Yasuda

Tokyo Metropolitan University
1-1 Minamiosawa, Hachioji, Tokyo 192-0397, Japan
E-mail: tamu@tmu.ac.jp
[Received March 4, 2011; accepted July 15, 2011]

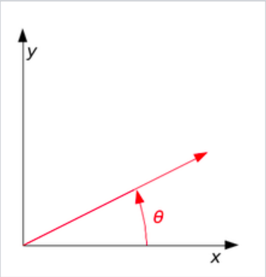
[4]

Rotation in $n-$ Dimensional Space

Informasi yang bisa diambil adalah sebagai berikut:

$$R^{(n)}(\theta_{1,2},\theta_{1,3},\ldots,\theta_{n,n-1}) := R^{(n)}_{n-1,n}(\theta_{n-1,n})$$
$$\times R^{(n)}_{n-2,n}(\theta_{n-2,n}) R^{(n)}_{n-2,n-1}(\theta_{n-2,n-1}) \times \cdots \times R^{(n)}_{2,n}(\theta_{2,n})$$
$$\times \cdots \times R^{(n)}_{2,3}(\theta_{2,3}) R^{(n)}_{1,n}(\theta_{1,n}) \times \cdots \times R^{(n)}_{1,3}(\theta_{1,3}) R^{(n)}_{1,2}(\theta_{1,2})$$
$$= \prod_{i=1}^{n-1}\left(\prod_{j=1}^{i} R^{(n)}_{n-i,n+1-j}(\theta_{n-i,n+1-j})\right) \quad \ldots\ldots (7)$$

Aljabar Linear: Rotasi pada 2 Dimensi



A counterclockwise rotation of a vector through angle θ. The vector is initially aligned with the x-axis.

Rotasi tidak mengubah norm vektor.

Konsekuensi

$$R^{(n)}(\theta_{1,2}, \theta_{1,3}, .., \theta_{n,n-1}) = \prod_{i=1}^{n-1} \left(\prod_{j=1}^{i} R^{(n)}_{n-i,n+1-j}(\theta_{n-i,n+1-j}) \right)$$

Menghasilkan matriks rotasi yang tidak mengubah norm vektor.

Key Take Points I

Pada rotasi pada 3-dimensi, saya temukan bahwa:

$$R^{(3)} = R23 \times R13 \times R12$$

Merupakan matriks yang *preserve norm vector*.

Sedangkan matriks $R^{(3)} = R23.R13.R12$ tidak memberikan rotasi tapi hanya memberikan kontraksi.

Paper III

Improved Black Hole Optimization Algorithm

Contents lists available at ScienceDirect
Journal of King Saud University – Computer and Information Sciences
journal homepage: www.sciencedirect.com

Improved Black Hole optimization algorithm for data clustering

Hasan Deeb [a], Archana Sarangi [a,*], Debahuti Mishra [a], Shubhendu Kumar Sarangi [b]
[a] Department of Computer Science Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India
[b] Department of Electronics & Instrumentation Engineering, Siksha 'O' Anusandhan Deemed to be University, Bhubaneswar, Odisha, India

[5]

REFERENCES

REFERENCES I

[1]
Hillier F S and Lieberman G J 2001 *Introduction to operations research* (New York, US: McGraw Hill)

[2]
Chachuat B 2011 MILP: Branch-and-bound search

[3]
Kania A and Sidarto K A 2016 Solving mixed integer nonlinear programming problems using spiral dynamics optimization algorithm

REFERENCES II

[4]
Tamura K and Yasuda K 2011 Spiral dynamics inspired optimization

[5]
Deeb H, Sarangi A, Mishra D and Sarangi S K 2020 Improved black hole optimization algorithm for data clustering