

PENGGUNAAN *LINEAR PROGRAMMING* DAN *GENETIC ALGORITHMS* DALAM MENYELESAIKAN *TIME-COST TRADE-OFF PROBLEM*

Andy Septianus¹, Danny Laorent², Paulus Nugraha³

ABSTRAK : Pada proyek konstruksi, hubungan *schedule* dan *budget* memiliki peranan yang sangat penting dalam menentukan kesuksesan proyek karena dari kedua faktor tersebut terjadi hubungan *time-cost trade-off* dari *direct cost* dan *indirect cost*. Dalam hal ini jika hubungan tersebut dapat dioptimalkan dengan meminimalkan *project's total cost* saat mereduksi durasi aktivitas maka biaya proyek akan berkurang secara signifikan.

Alasan ini yang membuat *schedule compression* menjadi penting untuk dikembangkan, beberapa metode diantaranya yang dapat digunakan adalah *Linear Programming* dan *Genetic Algorithms*. Dalam penelitian ini *scheduling compression* diselesaikan dengan *spreadsheet* dimana perhitungan metode *Linear Programming* dibantu dengan program *add-in* Jensen LP/IP sedangkan *Genetic Algorithms* dibantu dengan *Evolver*.

Dari proses dan hasil penelitian yang dilakukan diperoleh perbandingan kelemahan dan kelebihan kedua metode tersebut. *Linear Programming* relatif susah untuk digunakan karena harus mengubah *AOA* menjadi persamaan matematika dan untuk hubungan selain *Finish-Start* tanpa *lag/lead* diperlukan modifikasi *AOA* yang rumit. Selain itu juga LP hanya terbatas pada *linear activity time cost relation* saja. Meskipun demikian hasil yang diberikan oleh LP sangat memuaskan dan optimal.

Sedangkan *Genetic Algorithms* jauh lebih mudah digunakan karena dasarnya yang sederhana dan juga logika *scheduling* telah diterjemahkan menjadi *formula spreadsheet* yang berlaku untuk segala kondisi *predecessors*. Namun hasil yang diberikan oleh GA tidak sebaik LP karena hanya menghasilkan *approximate result*.

KATA KUNCI : *schedule compression, time-cost trade-off, linear programming, genetic algorithms*

1. PENDAHULUAN

Dunia konstruksi dikenal dengan permasalahannya yang sangat kompleks dan tingkat ketidakpastian yang tinggi. Hal ini hampir terjadi pada semua proyek konstruksi sehingga menyulitkan manajer proyek untuk menyelesaikan proyek sesuai *schedule* dan *budget* yang direncanakan. Berdasarkan kondisi tersebut maka dapat dipahami mengapa *delay* dan *cost overruns* sering terjadi pada proyek konstruksi sehingga *schedule compression* harus dilakukan. Namun mengingat adanya *time-cost trade-off* yang terjadi selama proses reduksi durasi aktivitas maka diperlukan metode pemilihan yang efektif dan optimum sehingga dapat dicapai *minimal total cost*. Dalam penelitian ini metode yang digunakan adalah *Linear Programming* dan *Genetic Algorithms* dengan bantuan *spreadsheet*.

¹Mahasiswa Universitas Kristen Petra, andy.septianus91@yahoo.com

²Mahasiswa Universitas Kristen Petra, d.laorent_zen@yahoo.com

³Dosen Universitas Kristen Petra, pnugraha@petra.ac.id

2. LANDASAN TEORI

2.1. Project Acceleration

Sebuah proyek konstruksi terdiri dari 2 fase utama yaitu *design* dan konstruksi, kedua fase ini sama pentingnya berkontribusi terhadap kesuksesan proyek. Sementara itu reduksi durasi proyek bisa dilakukan dalam 2 level : *strategic level* dan *tactical level*. *Strategic level* pada umumnya diterapkan sebelum proyek masuk fase konstruksi, sedangkan *tactical level* cenderung dilakukan saat fase konstruksi berjalan dan dalam penelitian ini dilakukan pada *tactical level*.

2.2. Method of Schedule Compression

Dalam penerapannya di lapangan, *schedule compression* pada *tactical level* memerlukan perhatian ekstra dan penyesuaian. Karena pada dasarnya kondisi yang terjadi selalu berbeda maka solusi yang dipilih pun bermacam-macam. Beberapa cara tradisional yang sering diadaptasi untuk melakukan *schedule compression* ialah (Awad S.Hanna Et al., 2005) : *overtime*, *shift work*, dan *overmanning*.

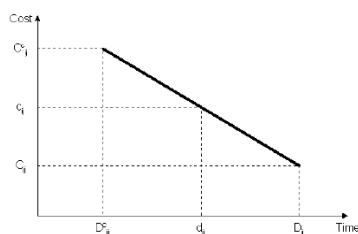
2.3. Time-Cost Trade-Off

Analisis *time-cost trade-off* dilakukan untuk mencapai keseimbangan yang baik antara *overall project cost*, *respective duration* dan juga mencapai *targeted milestones* yang diberikan *owner* atau mengganti waktu yang hilang akibat *delay* yang terjadi di lapangan (Roofigari-Esfahan, 2011b). Hubungan seperti terlihat pada **Gambar 1** ini yang lalu dideskripsikan sebagai *project time cost relation*.

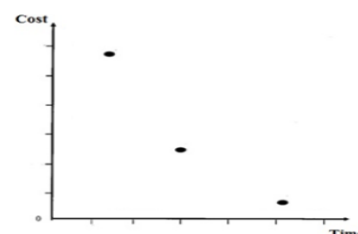


Gambar 1. Project Time-Cost Relation

Grafik ini diperoleh dari analisa *time-cost trade-off* pada setiap waktu penyelesaian proyek dari *completion time* terpendek hingga terpanjang. Respon perubahan biaya yang terjadi diperoleh dari *activity time cost relation* pada masing-masing aktivitas, dimana dua diantaranya yang dikenal adalah *linear activity time cost relation* (**Gambar 2**) dan *discrete activity time cost relation* (**Gambar 3**).



Gambar 2. Linear Activity Time Cost Relation



Gambar 3. Discrete Activity Time Cost Relation

2.4. Optimization Methods

Metode yang digunakan dalam menyelesaikan TCTO *problem* dapat dikategorikan dalam **Tabel 1**

Tabel 1. Optimization Methods

Mathematical Programming	Artificial Intelligence	Heuristic Method
Linear Programming (LP)	Genetic Algorithms	Prager's Structural Model
Integer Programming (IP)	Particle Swarm Optimization (PSO)	Moselhi's Structural Model
Dynamic Programming	Ant Colony Optimization (ACO)	Iterative Crashing Process
LP/IP Hybrid		Maximal Flow Minimal Cut

Alasan pemilihan *Linear Programming* karena merupakan salah satu metode optimisasi yang tertua, metode penyelesaian dengan pendekatan matematis dengan hasil optimal dan sederhana. Sedangkan alasan pemilihan *Genetic Algorithms* karena merupakan metode *AI* yang telah lama dikembangkan dan

memberikan hasil pendekatan yang memuaskan serta paling populer di antara metode AI lainnya sehingga aplikasi penyelesaian berbasis GAs relatif lebih mudah didapatkan, dipahami dan digunakan.

3. METODOLOGI PENELITIAN

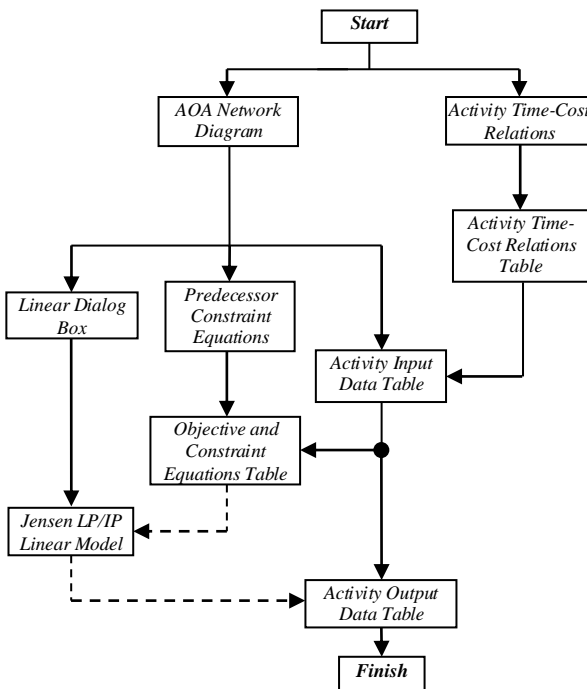
Program *excel add-in* yang digunakan ada 2 jenis yaitu Jensen LP/IP (LP) dan Evolver (GAs). Kemudian kedua metode tersebut digunakan menyelesaikan 4 *scheduling network* :

1. *Simple network* dengan hubungan *linear activity time-cost relation*
2. *Complex network* dengan hubungan *linear activity time-cost relation*
3. *Simple Network* dengan *discrete activity time-cost relation* (hanya dikerjakan dengan Evolver)
4. *Complex network with SS, SF, FF, lag or/and lead* dengan *linear activity time-cost relation*

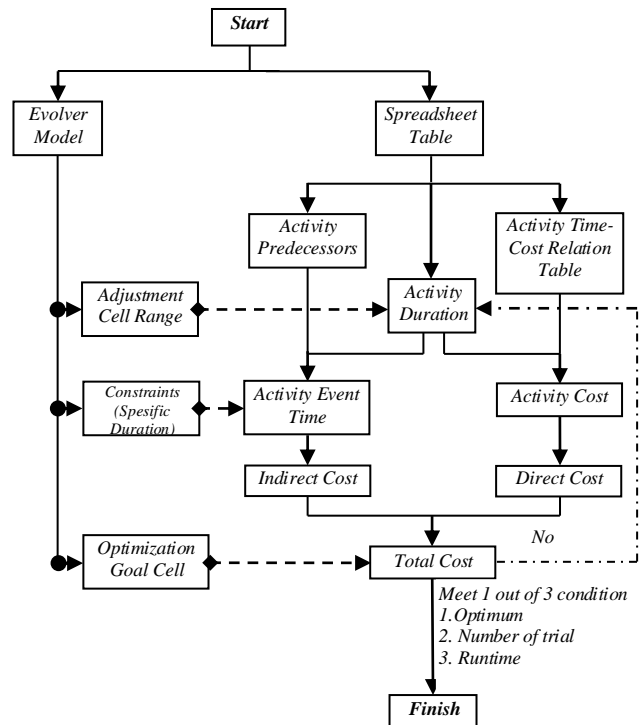
Simulasi pertama untuk memperoleh durasi optimum dengan *least cost* tanpa *deadline* sedangkan simulasi kedua dilakukan dengan merubah target durasi dari minimum sampai maksimum yang diizinkan. Hasil yang diperoleh berupa *direct cost* dan *indirect cost* untuk setiap durasi. Berikutnya ialah membuat *project time-cost relation graphic* dari titik-titik tersebut. Kemudian penyajian akhir diberikan dalam bentuk *ganttt chart* supaya mudah dilihat dan dimengerti.

4. ANALISA DAN PEMBAHASAN

Alur berpikir dalam proses pembuatan program *Linear Programming* dan *Genetic Algorithms* dapat dilihat pada **Gambar 4** dan **Gambar 5**



Gambar 4. Flowchart Program LP



Gambar 5. Flowchart Program GAs

Untuk *Linear Programming*, *scheduling network* diubah menjadi persamaan matematika yang terdiri dari 3 jenis persamaan yaitu :

1. *Objective Equation*

$$\text{Min total cost; } Z = \sum C_i \cdot Y_i + C_{ic} \cdot X_n + C_{nr}$$

Dimana,

C_i = reduction cost atau cost slope untuk setiap aktivitas (\$/days)

Y_i = reduction time untuk setiap aktivitas (days)

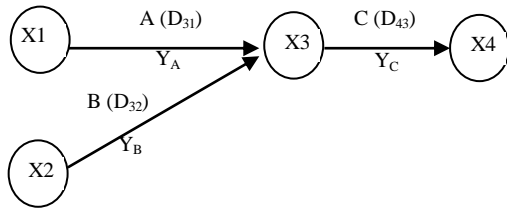
C_{ic} = indirect cost per day (\$/days)

X_n = event time pada node terakhir dari AOA network diagram (days)

C_{nr} = total normal cost sebelum dilakukan schedule compression (\$)

2. Constraints equation

Yang diperoleh dari mentransformasikan AOA (**Gambar 6**) menjadi pertidaksamaan *predecessors* (**Tabel 2**). Pertidaksamaan yang terbentuk mewakili durasi awal, durasi yang direduksi, urutan dan jenis hubungan yang terjadi antar aktivitas.



Gambar 6. AOA Network Diagram (FS)

A = activity's ID

X_2 = node akhir dari arrow/ aktivitas yang ditinjau

D_{31} = durasi awal dari aktivitas yang ditinjau

Y_A = reduction time dari aktivitas yang ditinjau

X_1 = node awal dari arrow/ aktivitas yang ditinjau

Tabel 2. Constraints Equations

$X_3 \geq D_{31} - Y_A + X_1$	\Rightarrow	$Y_A - X_1 + X_3 \geq D_{31}$
$X_3 \geq D_{32} - Y_B + X_2$	\Rightarrow	$Y_B - X_2 + X_3 \geq D_{32}$
$X_4 \geq D_{43} - Y_C + X_3$	\Rightarrow	$Y_C - X_3 + X_4 \geq D_{43}$

Tabel 3. Lower and Upper Bound

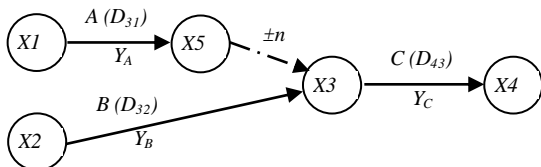
No.	Reduction Time	
	Lower Bound	Upper Bound
1	$Y_A \geq 0$	$Y_A \leq n_1$
2	$Y_B \geq 0$	$Y_B \leq n_2$
3	$Y_C \geq 0$	$Y_C \leq n_3$

3. Lower and upper bound

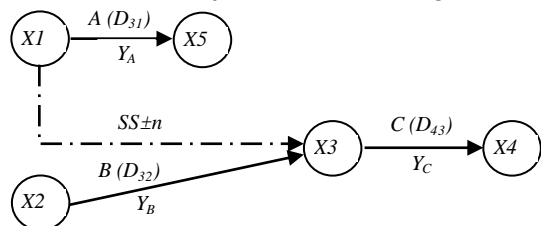
Batas atas dan batas bawah dari durasi aktivitas (Y_i) yang diizinkan untuk direduksi. Nilai *lower bound* selalu 0 sedangkan nilai *upper bound* didapat dari *crash range* atau *normal duration* dikurangi *maximal crash duration* (lihat **Tabel 3**)

Modified AOA

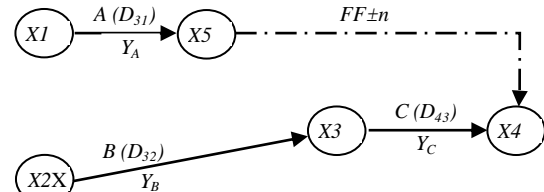
Hubungan *predecessor* yang dapat diwakili oleh AOA hanya jenis *Finish-Start* tanpa *lag/lead* sehingga untuk menyelesaikan *network* dengan jenis hubungan lain diperlukan cara tambahan yakni memodifikasi AOA tersebut (lihat **Gambar 7-10**) untuk kebutuhan membentuk *constraints equations*.



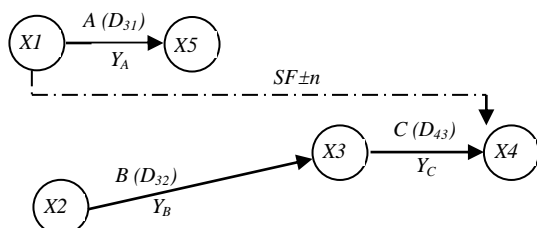
Gambar 7. Modified AOA untuk Lag/Lead



Gambar 9. Modified AOA untuk Hubungan SS



Gambar 8. Modified AOA untuk Hubungan FF



Gambar 10. Modified AOA untuk Hubungan SF

Untuk *Genetic Algorithms* ada 3 jenis *cell* yang dideskripsikan untuk mewakili *scheduling network*

1. *Optimization Cell*

Goal dari *optimization* dari TCTO problem adalah *minimize total cost*

2. *Adjustment Range Cell*

Cell range yang menentukan hasil *optimization cell* yakni durasi aktivitas. *Cell* ini diganti secara berulang kali hingga salah satu dari 3 kondisi tercapai : optimum solution, *number of trial* dan *runtime*

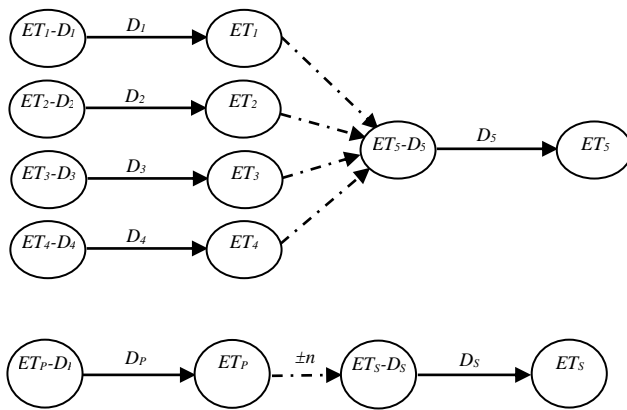
3. Constraints

Dalam *scheduling compression* segala hubungan *predecessor* telah diterjemahkan dalam *formula spreadsheet* pada perhitungan *event time* sehingga tidak ada lagi constraint yang diperlukan.

Namun untuk tujuan memperoleh *project time-cost relation graphic* diperlukan solusi jawaban optimum untuk setiap TCTO problem pada durasi tertentu sehingga ada tambahan *deadline constraint*.

Event Time

Hubungan antar aktivitas didefinisikan pada perhitungan *event time* (**Gambar 11**) dimana untuk masing-masing jenis hubungan *predecessor* berlaku rumus sebagai berikut (lihat **Tabel 4**):



Gambar 11. Ilustrasi Perhitungan Event Time (FS)

Tabel 4. Rumus Perhitungan Event Time

$ET_5 = \max (ET_{15}, ET_{25}, ET_{35}, ET_{45}, D_5)$	
Hubungan	Rumus Perhitungan
FS	$ET_S = (ET_P + D_S) \pm n$
SS	$ET_S = (ET_P - D_P + D_S) \pm n$
FF	$ET_S = (ET_P) \pm n$
SF	$ET_S = (ET_P - D_P) \pm n$

Dimana,

ET_P = Event Time of Precedence

ET_S = Event Time of Successor

D_P = Duration of Precedence

D_S = Duration of Successor

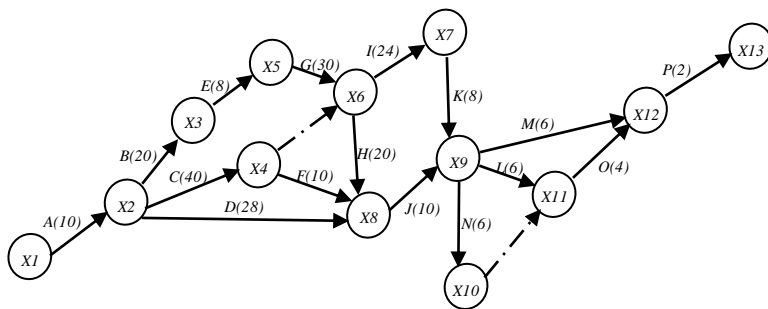
(+/-) n = lag/ lead

Simulasi

Dalam simulasi ini digunakan sebuah *complex network* dengan hubungan *linear activity time-cost relation* yang terdiri dari 16 aktivitas dan biaya tak langsung per hari \$120/hari.

Kemudian dari hubungan biaya yang ada dicari *optimum duration with minimal total cost* dan *minimum total cost with spesific duration* dengan 2 metode : LP dan GAs.

1. *Linear Programming*



Gambar 12. AOA Network dari simulasi

Network disamping (**Gambar12**) diubah ke dalam persamaan matematika yang mewakili hubungan (**Tabel 5**) dan batas reduksi (**Tabel 6**).

Kemudian diselesaikan dengan Jensen LP/IP sesuai persamaan obyektif yang telah dijelaskan.

Tabel 5. Constraints Equation

No.	Predecessor Constraint	No.	Predecessor Constraint	No.	Predecessor Constraint
1	$Y_A - X_1 + X_2 \geq 10$	7	$Y_I - X_6 + X_7 \geq 24$	13	$Y_N - X_9 + X_{10} \geq 6$
2	$Y_B - X_2 + X_3 \geq 20$	8	$Y_H - X_6 + X_8 \geq 20$	14	$Y_L - X_9 + X_{11} \geq 6$
3	$Y_C - X_2 + X_4 \geq 40$	9	$Y_F - X_4 + X_8 \geq 10$	15	$-X_{10} + X_{11} \geq 0$
4	$Y_E - X_3 + X_5 \geq 8$	10	$Y_D - X_2 + X_8 \geq 28$	16	$Y_M - X_9 + X_{12} \geq 6$
5	$Y_G - X_5 + X_6 \geq 30$	11	$Y_K - X_7 + X_9 \geq 8$	17	$Y_o - X_{11} + X_{12} \geq 4$
6	$-X_4 + X_6 \geq 0$	12	$Y_J - X_8 + X_9 \geq 10$	18	$Y_p - X_{12} + X_{13} \geq 2$

Tabel 6. Lower and Upper Bound

No.	Reduction Time		No.	Reduction Time		No.	Reduction Time		No.	Reduction Time	
	Lower Bound	Upper Bound		Lower Bound	Upper Bound		Lower Bound	Upper Bound		Lower Bound	Upper Bound
1	$Y_A \geq 0$	$Y_A \leq 0$	5	$Y_E \geq 0$	$Y_E \leq 0$	9	$Y_I \geq 0$	$Y_I \leq 10$	13	$Y_M \geq 0$	$Y_M \leq 3$
2	$Y_B \geq 0$	$Y_B \leq 0$	6	$Y_F \geq 0$	$Y_F \leq 4$	10	$Y_J \geq 0$	$Y_J \leq 4$	14	$Y_N \geq 0$	$Y_N \leq 2$
3	$Y_C \geq 0$	$Y_C \leq 0$	7	$Y_G \geq 0$	$Y_G \leq 20$	11	$Y_K \geq 0$	$Y_K \leq 1$	15	$Y_O \geq 0$	$Y_O \leq 0$
4	$Y_D \geq 0$	$Y_D \leq 8$	8	$Y_H \geq 0$	$Y_H \leq 12$	12	$Y_L \geq 0$	$Y_L \leq 1$	16	$Y_P \geq 0$	$Y_P \leq 0$

2. Genetic Algorithms

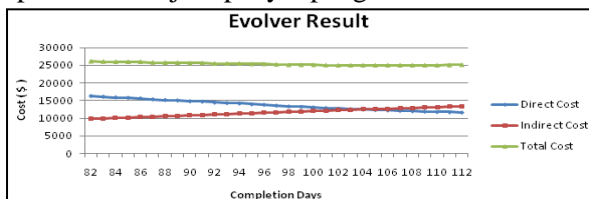
Pada GAS hanya diperlukan meng-input data-data yang ada seperti pada **Tabel 7** kemudian program tersebut dapat dijalankan.

Tabel 7. Spreadsheet untuk Evolver

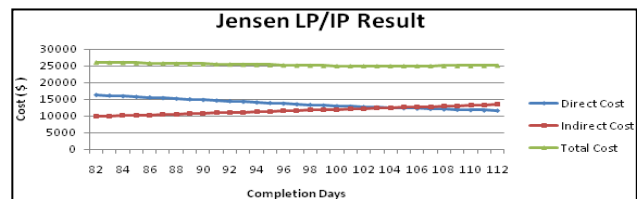
Act	Act ID	Predecessors/relation						Norm Dur	Crash Dur	Normal Cost	Crash Cost	Dur	Cost	Event Time
A	1							10	10	\$200	\$200	10	200	10
B	2	1						20	20	\$200	\$200	20	200	30
C	3	1						40	40	\$1,800	\$1,800	40	1800	50
D	4	1						28	20	\$500	\$580	28	500	38
E	5	2						8	8	\$150	\$150	8	150	38
F	6	3						10	6	\$100	\$260	10	100	60
G	7	5						30	10	\$3,000	\$6,600	30	3000	68
H	8	3	7					20	8	2,800	\$3,400	12	3200	80
I	9	3	7					24	14	1,000	\$1,650	14	1650	82
J	10	4	6	8				10	6	\$200	\$520	10	200	90
K	11	9						8	7	\$520	\$600	8	520	90
L	12	10	11					6	5	\$420	\$480	6	420	96
M	13	10	11					6	3	\$200	\$320	6	200	96
N	14	10	11					6	4	\$150	\$290	6	150	96
O	15	12	14					4	4	\$300	\$300	4	300	100
P	16	13	15					2	2	\$180	\$180	2	180	102
Total												12,770	102	

Hasil Simulasi

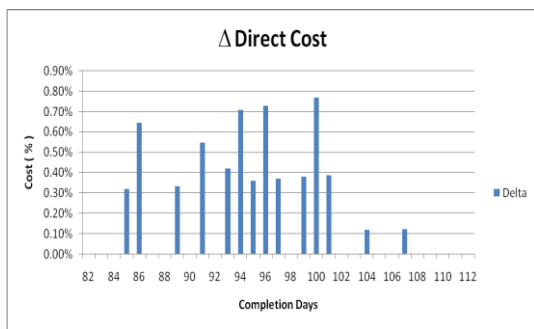
Dari simulasi Evolver (**Gambar 13**) dan Jensen LP/IP (**Gambar 14**) diperoleh hasil yang sama yakni durasi optimum 102 hari dengan *direct cost* \$12,770, *indirect cost* 12,240 dan *total cost* \$25,010. Namun untuk *minimize total cost with spesific duration* hasil dari keduanya sedikit berbeda karena pada GAS terjadi penyimpangan.



Gambar 13. Evolver Result



Gambar 14. Jensen LP/IP Result



Perbedaan yang terjadi pada *network* kedua ini kurang dari 1% (lihat **Gambar 15**) sedangkan pada *network* pertama (6 aktivitas) tidak terjadi deviasi dan pada *network* keempat (23 aktivitas) terjadi deviasi kurang dari 0.06%. Hal ini menyatakan bahwa nilai deviasi yang terjadi bersifat relatif karena sangat dipengaruhi oleh *activity time-cost relation* dan *network* yang dikompres. Namun secara umum GAs sudah memberikan hasil yang sudah sangat mendekati optimum dengan tingkat kesalahan yang relatif kecil dibandingkan LP.

Gambar 15. *Direct Cost Deviation* antara GAs dan LP

5. KESIMPULAN DAN SARAN

Kesimpulan dari penelitian tentang penggunaan *Linear Programming* dan *Genetic Algorithms*:

1. *Linear Programming*

➤ Kelebihan

- Memberikan hasil yang *absolute* dan *optimal*.
- Lebih cepat mengetahui *answer feasibility*.

➤ Kekurangan

- Tidak dapat menyelesaikan *discrete activity time-cost relation*.
- Susah mentransformasikan *scheduling network* menjadi persamaan matematika.
- Harus menggambar AOA sebelum mendeskripsikan *activities dependencies*.
- Membutuhkan waktu yang lama untuk pengguna tanpa dasar pengetahuan LP.
- Program jadi tidak dapat langsung digunakan karena banyak persamaan harus diganti.
- Terbatas untuk beberapa aktivitas karena kesulitan menggambar AOA dan persamaannya.
- Untuk hubungan selain *FS* tanpa *lag/lead* perlu 2 kali gambar AOA.
- Hanya bisa memberikan satu solusi (tidak ada alternatif lain).

2. *Genetic Algorithms*

➤ Kelebihan

- Dapat menyelesaikan semua jenis *activity time-cost relation*.
- Program yang dibuat berdasarkan AON *network diagram* sehingga lebih mudah dimengerti.
- Tidak diperlukan untuk menggambar AON *network diagram*.
- Lebih mudah untuk dikembangkan terhadap berbagai variasi permasalahan *scheduling*.
- Mampu menyelesaikan *TCTO problems* dengan banyak aktivitas.
- Alternatif jawaban lain yang telah dicoba dapat dilihat pada *log/report*.

➤ Kekurangan

- Membutuhkan banyak waktu dalam menjalankan proses *iterasi*.
- Tidak dapat memastikan apakah optimum jika berhenti akibat *run-out time /member of trial*.
- Hanya memberikan *approximate result*.
- Tidak memberikan hasil yang stabil karena tergantung durasi awal yang dimasukkan.

6. DAFTAR PUSTAKA

- Awad S. Hanna, Craig S. Taylor, and Sullivan, K. T. (2005). Impact of Extended Overtime on Construction Labor Productivity. *Journal of Construction Engineering and Management*, 131(6), 734-739
- Roofigari-esfahan, N. (2011b). *Using the Analytical Hierarchy Process in Compressing Schedules of Construction Projects*, 1–6.