



Perancangan Ulang Tata Letak Fasilitas Produksi Menggunakan *Modified Squirrel Search Algorithm*

Selsa Ulfia Siswanti, Annisa Kesy Garside*, Ikhlasul Amalynda

Program Studi Teknik Industri, Universitas Muhammadiyah Malang, I. Raya Tlogomas No 246, Malang, 65144, Indonesia

INFORMASI ARTIKEL

Sejarah Artikel:

Artikel Masuk: 14 Agustus 2023

Artikel direvisi: 11 Oktober 2023

Artikel diterima: 12 November 2023

Kata kunci

Jarak Perpindahan Material
Metaheuristik
Squirrel Search Algorithm
Tata Letak Fasilitas

Keywords

Material Handling Distance
Metaheuristics
Squirrel Search Algorithm
Facility Layout.

ABSTRAK

CV XYZ merupakan sebuah perusahaan yang memproduksi cup gelas dan galon. Saat ini, penempatan beberapa departemen CV XYZ masih belum sesuai dengan aliran proses produksi sehingga terjadi gerakan balik (*backtracking*) yang mengakibatkan jarak perpindahan material lebih panjang. Penelitian ini bertujuan untuk merancang ulang tata letak departemen sehingga jarak perpindahan material menjadi lebih pendek. Metode yang diusulkan adalah *modified Squirrel Search Algorithm* (SSA). Berdasarkan data layout perusahaan saat ini, dimensi setiap departemen dan frekuensi perpindahan material diperoleh 9 alternatif tata letak (*layout*) usulan. Layout dengan urutan departemen A-B-E-G-F-C-D-H merupakan solusi terbaik karena menghasilkan total jarak perpindahan material paling minimum sebesar 2731,5 m. Total jarak perpindahan material berkurang sebesar 372 m dibandingkan dengan total jarak perpindahan material pada layout perusahaan. Sehingga diperoleh efisiensi tata letak usulan dengan algoritma SSA sebesar 11,99%.

ABSTRACT

CV XYZ is a company that manufactures disposable cups and water gallons. Currently, several departments within CV XYZ differ from the production process flow, resulting in *backtracking* movements that lead to longer material handling distances. This research aims to redesign the department layout to minimize material handling distances. The proposed method is the *modified Squirrel Search Algorithm* (SSA). Nine alternative layout proposals were generated based on the current company layout data, each department's dimensions, and the material handling frequency. The layout with the sequence of departments A-B-E-G-F-C-D-H is the best solution, resulting in the minimum total material handling distance of 2731.5 m. It represents a reduction of 372 m compared to the total material handling distance in the current company layout. Therefore, the proposed layout efficiency with the SSA algorithm is 11.99%.

* Penulis Korespondensi

Annisa Kesy Garside
E-mail: annisa@umm.ac.id

This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



© 2023. Some rights reserved

1. PENDAHULUAN

Tata letak fasilitas (*facility layout*) dapat didefinisikan sebagai tata cara pengaturan fasilitas-fasilitas pabrik guna menunjang kelancaran proses produksi (Wignjosebroto, 2009). Tata letak fasilitas secara langsung mempengaruhi kinerja sistem manufaktur, seperti logistik, aliran informasi, kapasitas produksi, efisiensi peralatan dan pekerjaan, biaya produksi serta keselamatan pekerja (Tang et al., 2022). Salah satu tujuan perencanaan tata letak fasilitas adalah meminimasi jarak dan biaya perpindahan material. Tata letak fasilitas dapat mempengaruhi biaya perpindahan material sekitar 20-70% dari total biaya operasi. Hal tersebut dapat diartikan bahwa untuk menekan biaya produksi dapat dilakukan dengan cara menurunkan biaya perpindahan material (Rajak, 2018).

Dalam beberapa tahun terakhir, banyak penelitian tentang optimasi tata letak telah dilakukan dengan menggunakan algoritma metaheuristik (Prayogo et al.,

2018). Algoritma metaheuristik yang diusulkan untuk menyelesaikan tata letak fasilitas diantaranya algoritma *Particle Swarm Optimization* (Liu & Liu, 2019), *Whale Optimization Algorithm* (Kaveh & Vazirinia, 2019), *Biogeography Based Optimization* (Singh & Ingole, 2019), *Salp Swarm Algorithm* (Elkassas & ElWakil, 2019), *Greedy Randomized Adaptive Search Procedure* (Cravo & Amaral, 2019), *Hybrid Coral Reefs Optimization* (Garcia-Hernandez et al., 2020), *Graphic Processing Units* (Sun et al., 2022), *Differential Evolution* (Krömer et al., 2020), *Evolutionary Algorithm* (Senatra & Widyadana, 2022), *Simulated Annealing* (Kusumaningsih et al., 2022), dan *Memetic Algorithm* (Wang & Xu, 2023).

Pada penelitian yang dilakukan oleh Elkassas & ElWakil (2019), *Salp Swarm Algorithm* menghasilkan solusi yang lebih baik dibanding *Particle Swarm Optimization*. Penelitian lain dilakukan oleh Tongur et al. (2020) dengan membandingkan algoritma *Migrating*



Birds Optimization, Simulated Annealing, dan Tabu Search. Hasil penelitian menunjukkan bahwa *Migrating Birds Optimization dan Simulated Annealing* menghasilkan solusi lebih baik dibanding *Tabu Search*. Al-Saati & Aslgamal (2022) memodifikasi algoritma *Squirrel Search Algorithm (SSA)* dengan memperkenalkan operator baru untuk menyelesaikan *single-row facility layout problems*. Solusi *modified SSA* mampu mengungguli algoritma *Semidefinite Programming (SDP)* dan *Cuckoo Optimization Algorithm (COA)*.

CV XYZ merupakan sebuah perusahaan manufaktur yang memproduksi gelas cup dan galon dengan berbagai macam bentuk, volume dan berat sesuai dengan permintaan pelanggan. Perusahaan memiliki 8 departemen yaitu gudang bahan baku, *crusher, mixer, extruder, thermoforming, blow moulding, packing* dan gudang barang jadi. Berdasarkan penempatan departemen-departemen pada CV XYZ masuk dalam kategori *two-row facility layout problem*. Dari hasil pengamatan, penempatan beberapa departemen masih belum sesuai dengan aliran proses produksi. Sehingga terjadi gerakan balik (*backtracking*) dari departemen *packing* ke gudang barang jadi. Selain itu, sisa aval dari departemen *thermoforming* yang akan dipindahkan ke departemen *crusher* harus melewati departemen *packing* terlebih dahulu. Dengan adanya gerakan balik tersebut menyebabkan jarak perpindahan material menjadi lebih panjang.

Setiap algoritma metaheuristik mampu menghasilkan performansi yang berbeda untuk permasalahan tata letak fasilitas (Prayogo et al., 2018). Penelitian oleh Al-Saati & Aslgamal (2022) mengusulkan modified *Squirrel Search Algorithm (SSA)* untuk menyelesaikan *single-row facility layout problem*, dimana solusi yang dihasilkan lebih baik dibanding SDP dan COA. Berdasarkan permasalahan yang dihadapi perusahaan, penelitian ini bertujuan untuk merancang ulang tata letak fasilitas CV XYZ dengan menggunakan modified SSA. Dalam penelitian ini, *modified SSA* yang diusulkan oleh Al-Saati & Aslgamal (2022) akan disesuaikan untuk menyelesaikan *two-row facility layout problem*.

2. METODE PENELITIAN

2.1. Squirrel Search Algorithm (SSA)

Squirrel Search Algorithm (SSA) merupakan sebuah algoritma yang terinspirasi dari seekor tupai yang mencari makan di antara pepohonan untuk memenuhi kebutuhannya (Al-Saati & Algamal, 2022). Tahap awal dalam menyelesaikan permasalahan tata letak fasilitas menggunakan SSA adalah inialisasi solusi. Dalam tahap ini, dilakukan penentuan parameter seperti populasi, iterasi, batas atas nilai acak (*Upper Bound*), batas bawah nilai acak (*Lower Bound*). Selanjutnya pembaruan lokasi tupai dengan menggunakan persamaan (1) sampai dengan persamaan (3). Tupai akan memperbarui posisi lokasi berdasarkan 3 faktor yaitu lokasi sebelumnya, jarak luncur dan probabilitas kedekatan predator.

a. Tupai terbang dari pohon *hickory nut* menuju pohon aren (*acorn tree*)

$$(S_{i,j}^{t+1})_a = \begin{cases} \text{if } R_1 \geq P_{pd}, & (S_{i,j}^t)_a + G_d \times C \times ((S_{i,j}^t)_h - (S_{i,j}^t)_a) \\ \text{Otherwise} & \text{Random location} \end{cases} \quad (1)$$

b. Tupai terbang dari pohon aren menuju pohon biasa (*normal tree*)

$$(S_{i,j}^{t+1})_n = \begin{cases} \text{if } R_2 \geq P_{pd}, & (S_{i,j}^t)_n + G_d \times C \times ((S_{i,j}^t)_a - (S_{i,j}^t)_n) \\ \text{Otherwise} & \text{Random location} \end{cases} \quad (2)$$

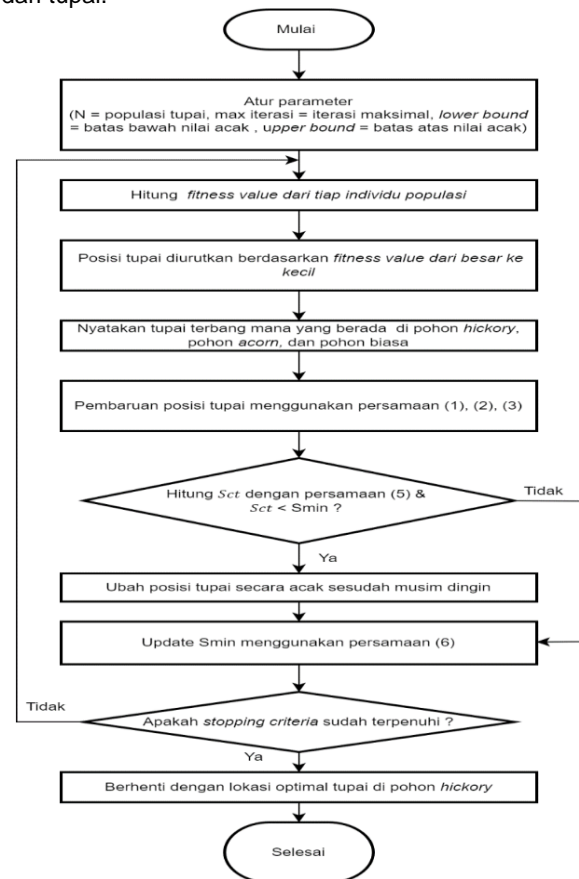
c. Tupai terbang dari pohon biasa menuju pohon *hickory nut*

$$(S_{i,j}^{t+1})_n = \begin{cases} \text{if } R_3 \geq P_{pd}, & (S_{i,j}^t)_n + G_d \times C \times ((S_{i,j}^t)_h - (S_{i,j}^t)_n) \\ \text{Otherwise} & \text{Random location} \end{cases} \quad (3)$$

$(S_{i,j}^t)_a$ menunjukkan lokasi tupai berada di pohon aren, $(S_{i,j}^t)_h$ menunjukkan lokasi tupai berada di pohon *hickory nut* dan $(S_{i,j}^t)_n$ menunjukkan lokasi tupai berada di pohon biasa. $R_1, R_2, \text{ dan } R_3$ merupakan angka yang dibuat secara acak $[0, 1]$. C merupakan koefisien luncur yang melintasi ruang pencarian. P_{bd} merupakan probabilitas bahwa predator sudah dekat. G_D merupakan konstanta jarak luncur yang dihitung menggunakan persamaan (4).

$$G_D = \frac{H_{loss}}{ScF \cdot \tan(\varphi)} \quad (4)$$

dimana H_{loss} merupakan ketinggian setelah meluncur 8m, ScF merupakan faktor penskalaan yang ditetapkan antara 16 dan 37 dan φ yang menunjukkan sudut jatuh dari tupai.



Gambar 1. Flowchart SSA

Setelah pembaruan lokasi posisi, tupai akan mengamati terjadinya musim dingin. Nilai minimum dari konstanta musim dingin dapat dihitung menggunakan persamaan (5).

$$S_c^i = \sqrt{\sum_{k=1}^d (S_{a,k}^i - S_{h,k}^i)^2} \quad (5)$$

$$S_{min} = \frac{10E^{-6}}{365^{t/(tm/2,5)}} \quad (6)$$

Hasil dari konstanta pada persamaan (5) akan dibandingkan dengan nilai S_{min} . Jika nilai $S_c^i < S_{min}$ dapat diartikan musim dingin telah berakhir, tupai terbang yang tidak dapat menemukan makanan untuk musim dingin akan diubah posisinya secara acak. Gambar 1 menunjukkan flowchart algoritma SSA.

2.2. Tahapan Perancangan Tata Letak Fasilitas Menggunakan Modified SSA

Langkah-langkah dalam perancangan tata letak fasilitas menggunakan *modified* SSA sebagai berikut:

1. Pengumpulan data
Data-data yang dibutuhkan dalam perancangan ulang tata letak fasilitas sebagai berikut:
 - (1) Proses produksi dan departemen- departemen pada CV XYZ.
 - (2) Dimensi panjang dan lebar setiap departemen. Dimensi tersebut digunakan dalam penentuan titik tengah setiap departemen dan perhitungan jarak antar dua departemen.
 - (3) *Layout* perusahaan pada kondisi saat ini.
 - (4) Frekuensi perpindahan material dari departemen satu ke departemen lainnya. Data ini diperoleh melalui pengamatan secara langsung.
2. Pembuatan *coding*
Coding dibuat dengan menggunakan MATLAB R2018 berdasarkan *flowchart Squirrel Search Algorithm* (SSA) yang telah disesuaikan dengan permasalahan tata letak fasilitas.
3. Uji verifikasi
Uji verifikasi dilakukan untuk menguji *coding* yang telah dibuat sebelumnya. Uji verifikasi dilakukan menggunakan kasus sederhana. Pengujian tersebut bertujuan untuk memastikan bahwa coding telah bebas *error*. Sehingga program dapat berjalan hingga menemukan solusi sesuai dengan algoritma yang digunakan. Jika *coding* masih belum bebas *error* maka dilakukan perbaikan *coding*.
4. Uji Validasi
Uji validasi dilakukan dengan membandingkan hasil perhitungan manual menggunakan Ms. Excel dengan perhitungan pada MATLAB menggunakan kasus sederhana. *Coding* SSA dikatakan valid jika hasil perhitungan manual dari kasus sederhana dengan hasil solusi dari SSA sama.
5. Perhitungan total jarak perpindahan material.
Persamaan (7) digunakan untuk menghitung total jarak perpindahan material (Guan et al., 2019).

$$D_{ij} = \sum_{i=1}^N \sum_{j=1}^N F_{ij} * d_{ij} \\ = \sum_{i=1}^N \sum_{j=1}^N F_{ij} * (|x_i - x_j| + |y_i - y_j|) \quad (7)$$

Dimana D_{ij} : total jarak perpindahan material; F_{ij} :

frekuensi aliran material dari departemen asal i ke departemen tujuan j ; d_{ij} : jarak antara departemen i ke departemen j ; (X_i, Y_i) : koordinat titik tengah dari departemen i ; dan (X_j, Y_j) : koordinat titik tengah departemen j .

Jarak antar departemen dihitung dengan metode *rectilinear*. Metode *rectilinear* sering digunakan karena mudah dimengerti. Selain itu metode ini lebih sesuai untuk penentuan jarak antar kota dan jarak antar fasilitas di mana peralatan pemindahan material hanya dapat bergerak secara lurus (Muharni et al., 2020).

6. Perancangan tata letak fasilitas menggunakan SSA
Berdasarkan *coding* yang telah terverifikasi dan tervalidasi, algoritma SSA digunakan untuk mencari alternatif usulan berupa urutan departemen. Pada saat menjalankan algoritma SSA, parameter berupa jumlah populasi dan jumlah iterasi ditetapkan oleh peneliti. Selanjutnya tata letak fasilitas dirancang ulang dengan mengacu urutan departemen yang dihasilkan dari algoritma SSA dan dimensi tiap departemen. Dari beberapa alternatif tata letak (*layout*) yang dihasilkan oleh SSA, *layout* yang dipilih sebagai *layout* usulan adalah *layout* yang memiliki total jarak perpindahan material paling minimum.
7. Perbandingan total jarak perpindahan material pada *layout* perusahaan dan *layout* usulan
Pada tahap ini dilakukan perbandingan total jarak perpindahan material pada *layout* perusahaan dan usulan. Berdasarkan hasil perbandingan ini dapat diketahui apakah algoritma *modified* SSA mampu menyelesaikan permasalahan CV. XYZ

3. HASIL DAN PEMBAHASAN

3.1. Proses Produksi

Produk yang dihasilkan oleh CV XYZ adalah *cup* gelas dan galon yang berbahan dasar polipropilena. Produk yang banyak dihasilkan adalah *cup* gelas untuk kemasan air minum. Proses produksi *cup* gelas melalui 5 tahapan sebagai berikut:

1. Penghancuran aval. Tahap proses penghancuran aval merupakan tahap dimana sisa material aval (lembaran plastik) akan dihancurkan menjadi kepingan plastik. Penghancuran aval ini dilakukan menggunakan mesin *crusher*.
2. Pencampuran. Sisa aval yang telah dihancurkan akan dicampur dengan biji plastik murni yaitu polypropylen. Selain menggunakan polipropilena sebagai bahan baku, terdapat bahan campuran yaitu masplene dan *Linear Low-Density Polyethylene* (LLDPE) yang digunakan agar *cup* gelas tidak mudah pecah dan transparan. Pencampuran dilakukan menggunakan mesin *mixer*.
3. Pembuatan aval. Setelah seluruh bahan tercampur, tahap selanjutnya adalah pembuatan aval menggunakan mesin *extruder*. Aval yang dihasilkan memiliki berat dan ketebalan yang berbeda sesuai dengan permintaan konsumen.
4. Pencetakan *cup* gelas. Aval yang telah digulung akan dicetak menjadi *cup* gelas menggunakan mesin *thermoforming* dengan bentuk dan volume sesuai dengan permintaan konsumen. Pada proses pencetakan *cup* gelas, sisa material aval akan

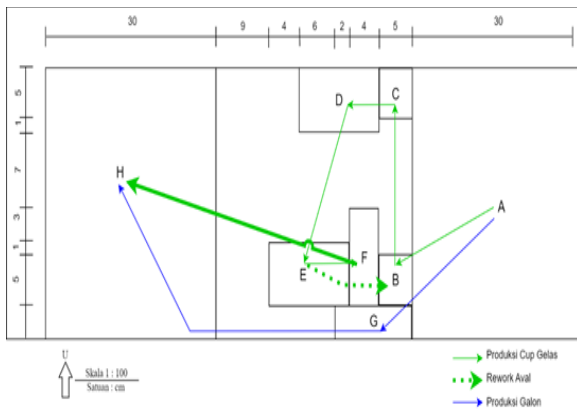
dihancurkan menggunakan mesin *crusher*.
 5. Pengemasan. Tahap ini *cup* gelas yang kualitasnya telah memenuhi standar akan dikemas dalam karton dan diberi label.

Selain memproduksi cup gelas, CV. XYZ juga memproduksi galon dengan ukuran 15 L. Produksi galon memiliki 2 tahapan sebagai berikut:

1. Pembuatan Galon. Proses ini menggunakan mesin *blow moulding*. Mesin ini akan meniup *preform* menjadi bentuk galon sesuai dengan cetakan mesin tersebut.
2. Pengemasan. Setelah keluar dari mesin *blow moulding*, galon akan langsung dikemas menggunakan plastik, agar bagian dalam galon tidak terkena debu saat berada di gudang.

3.2. Layout Kondisi Saat Ini

Gambar 2 menunjukkan *layout* perusahaan dan dimensi dari departemen pada Tabel 1. Sedangkan frekuensi perpindahan material yang terjadi tiap hari pada CV XYZ dapat dilihat pada Tabel 2.



Gambar 2. Kondisi Saat ini

Tabel 1. Dimensi Departemen

No.	Departemen	Kode	Panjang (m)	Lebar (m)
1.	Gudang Bahan Baku	A	30	25
2.	Crusher	B	5	5
3.	Mixer	C	5	5
4.	Extruder	D	12	6
5.	Thermoforming	E	12	6
6.	Packing	F	4	9
7.	Blow Moulding	G	9	3
8.	Gudang Barang Jadi	H	30	25

Tabel 2. Frekuensi Perpindahan Material

F/T	A	B	C	D	E	F	G	H
A	0	0	8	0	0	1	4	0
B	0	0	12	0	0	0	0	0
C	0	0	0	18	0	0	0	0
D	0	0	0	0	21	0	0	0
E	0	21	0	0	0	20	0	0
F	0	0	0	0	0	0	0	24
G	0	0	0	0	0	0	0	9
H	0	0	0	0	0	0	0	0

3.3. Perancangan Tata Letak Fasilitas

Gambar 3 menunjukkan *coding* yang telah dibuat sesuai dengan *flowchart* SSA. Langkah selanjutnya adalah melakukan uji verifikasi dan validasi. Data yang digunakan untuk uji validasi menggunakan kasus sederhana dengan 5 departemen.

```

1 function [BestFit, BestPositions, Best, Convergence_curve] = SSA(N, Max_iter, lb, ub, xi, yi, f);
2 %
3 % (N) = size(f);
4 %
5 % Max_iter=50; % maximum generations
6 % Nb=50; % Numbers of Squirrels
7 % Ndim=100;
8 % lb=2*zeros(1, dim); %
9 % ub=5*ones(1, dim);
10 % nfood=4; % Number of food resources
11 % food=1; % Sticky nut tree
12 % food=3; % Acorn nut tree
13 % food=4; % No food trees
14 %
15 % Pmax=1; % Maximum gliding distance
16 % Pmin=0; % Minimum gliding distance
17 % Awand(0, 1);
18 % r=rand(0, 1); % Update flying rate for each SSA
19 %
20 % Initializing arrays
21 % P=zeros(N, 1); % Frequency
22 % v=zeros(N, dim); % Velocities
23 %
24 % Initialize the population
25 % x=initialization(N, Max_iter, dim, lb, ub);
26 %
27 % Randomize the initial position for each squirrel
28 %
29 %
30 %
31 %
32 %
33 %
34 %
35 %
36 %
37 %
38 %
39 %
40 %
41 %
42 %
43 %
44 %
45 %
46 %
47 %
48 %
49 %
50 %
51 %
52 %
53 %
54 %
55 %
56 %
57 %
58 %
59 %
60 %
61 %
62 %
63 %
64 %
65 %
66 %
67 %
68 %
69 %
70 %
71 %
72 %
73 %
74 %
75 %
76 %
77 %
78 %
79 %
80 %
81 %
82 %
83 %
84 %
85 %
86 %
87 %
88 %
89 %
90 %
91 %
92 %
93 %
94 %
95 %
96 %
97 %
98 %
99 %
100 %
101 %
102 %
103 %
104 %
105 %
106 %
107 %
108 %
109 %
110 %
111 %
112 %
113 %
114 %
115 %
116 %
117 %
118 %
119 %
120 %
121 %
122 %
123 %
124 %
125 %
126 %
127 %
128 %
129 %
130 %
131 %
132 %
133 %
134 %
135 %
136 %
137 %
138 %
139 %
140 %
141 %
142 %
143 %
144 %
145 %
146 %
147 %
148 %
149 %
150 %
151 %
152 %
153 %
154 %
155 %
156 %
157 %
158 %
159 %
160 %
161 %
162 %
163 %
164 %
165 %
166 %
167 %
168 %
169 %
170 %
171 %
172 %
173 %
174 %
175 %
176 %
177 %
178 %
179 %
180 %
181 %
182 %
183 %
184 %
185 %
186 %
187 %
188 %
189 %
190 %
191 %
192 %
193 %
194 %
195 %
196 %
197 %
198 %
199 %
200 %
201 %
202 %
203 %
204 %
205 %
206 %
207 %
208 %
209 %
210 %
211 %
212 %
213 %
214 %
215 %
216 %
217 %
218 %
219 %
220 %
221 %
222 %
223 %
224 %
225 %
226 %
227 %
228 %
229 %
230 %
231 %
232 %
233 %
234 %
235 %
236 %
237 %
238 %
239 %
240 %
241 %
242 %
243 %
244 %
245 %
246 %
247 %
248 %
249 %
250 %
251 %
252 %
253 %
254 %
255 %
256 %
257 %
258 %
259 %
260 %
261 %
262 %
263 %
264 %
265 %
266 %
267 %
268 %
269 %
270 %
271 %
272 %
273 %
274 %
275 %
276 %
277 %
278 %
279 %
280 %
281 %
282 %
283 %
284 %
285 %
286 %
287 %
288 %
289 %
290 %
291 %
292 %
293 %
294 %
295 %
296 %
297 %
298 %
299 %
300 %
301 %
302 %
303 %
304 %
305 %
306 %
307 %
308 %
309 %
310 %
311 %
312 %
313 %
314 %
315 %
316 %
317 %
318 %
319 %
320 %
321 %
322 %
323 %
324 %
325 %
326 %
327 %
328 %
329 %
330 %
331 %
332 %
333 %
334 %
335 %
336 %
337 %
338 %
339 %
340 %
341 %
342 %
343 %
344 %
345 %
346 %
347 %
348 %
349 %
350 %
351 %
352 %
353 %
354 %
355 %
356 %
357 %
358 %
359 %
360 %
361 %
362 %
363 %
364 %
365 %
366 %
367 %
368 %
369 %
370 %
371 %
372 %
373 %
374 %
375 %
376 %
377 %
378 %
379 %
380 %
381 %
382 %
383 %
384 %
385 %
386 %
387 %
388 %
389 %
390 %
391 %
392 %
393 %
394 %
395 %
396 %
397 %
398 %
399 %
400 %
401 %
402 %
403 %
404 %
405 %
406 %
407 %
408 %
409 %
410 %
411 %
412 %
413 %
414 %
415 %
416 %
417 %
418 %
419 %
420 %
421 %
422 %
423 %
424 %
425 %
426 %
427 %
428 %
429 %
430 %
431 %
432 %
433 %
434 %
435 %
436 %
437 %
438 %
439 %
440 %
441 %
442 %
443 %
444 %
445 %
446 %
447 %
448 %
449 %
450 %
451 %
452 %
453 %
454 %
455 %
456 %
457 %
458 %
459 %
460 %
461 %
462 %
463 %
464 %
465 %
466 %
467 %
468 %
469 %
470 %
471 %
472 %
473 %
474 %
475 %
476 %
477 %
478 %
479 %
480 %
481 %
482 %
483 %
484 %
485 %
486 %
487 %
488 %
489 %
490 %
491 %
492 %
493 %
494 %
495 %
496 %
497 %
498 %
499 %
500 %
501 %
502 %
503 %
504 %
505 %
506 %
507 %
508 %
509 %
510 %
511 %
512 %
513 %
514 %
515 %
516 %
517 %
518 %
519 %
520 %
521 %
522 %
523 %
524 %
525 %
526 %
527 %
528 %
529 %
530 %
531 %
532 %
533 %
534 %
535 %
536 %
537 %
538 %
539 %
540 %
541 %
542 %
543 %
544 %
545 %
546 %
547 %
548 %
549 %
550 %
551 %
552 %
553 %
554 %
555 %
556 %
557 %
558 %
559 %
560 %
561 %
562 %
563 %
564 %
565 %
566 %
567 %
568 %
569 %
570 %
571 %
572 %
573 %
574 %
575 %
576 %
577 %
578 %
579 %
580 %
581 %
582 %
583 %
584 %
585 %
586 %
587 %
588 %
589 %
590 %
591 %
592 %
593 %
594 %
595 %
596 %
597 %
598 %
599 %
600 %
601 %
602 %
603 %
604 %
605 %
606 %
607 %
608 %
609 %
610 %
611 %
612 %
613 %
614 %
615 %
616 %
617 %
618 %
619 %
620 %
621 %
622 %
623 %
624 %
625 %
626 %
627 %
628 %
629 %
630 %
631 %
632 %
633 %
634 %
635 %
636 %
637 %
638 %
639 %
640 %
641 %
642 %
643 %
644 %
645 %
646 %
647 %
648 %
649 %
650 %
651 %
652 %
653 %
654 %
655 %
656 %
657 %
658 %
659 %
660 %
661 %
662 %
663 %
664 %
665 %
666 %
667 %
668 %
669 %
670 %
671 %
672 %
673 %
674 %
675 %
676 %
677 %
678 %
679 %
680 %
681 %
682 %
683 %
684 %
685 %
686 %
687 %
688 %
689 %
690 %
691 %
692 %
693 %
694 %
695 %
696 %
697 %
698 %
699 %
700 %
701 %
702 %
703 %
704 %
705 %
706 %
707 %
708 %
709 %
710 %
711 %
712 %
713 %
714 %
715 %
716 %
717 %
718 %
719 %
720 %
721 %
722 %
723 %
724 %
725 %
726 %
727 %
728 %
729 %
730 %
731 %
732 %
733 %
734 %
735 %
736 %
737 %
738 %
739 %
740 %
741 %
742 %
743 %
744 %
745 %
746 %
747 %
748 %
749 %
750 %
751 %
752 %
753 %
754 %
755 %
756 %
757 %
758 %
759 %
760 %
761 %
762 %
763 %
764 %
765 %
766 %
767 %
768 %
769 %
770 %
771 %
772 %
773 %
774 %
775 %
776 %
777 %
778 %
779 %
780 %
781 %
782 %
783 %
784 %
785 %
786 %
787 %
788 %
789 %
790 %
791 %
792 %
793 %
794 %
795 %
796 %
797 %
798 %
799 %
800 %
801 %
802 %
803 %
804 %
805 %
806 %
807 %
808 %
809 %
810 %
811 %
812 %
813 %
814 %
815 %
816 %
817 %
818 %
819 %
820 %
821 %
822 %
823 %
824 %
825 %
826 %
827 %
828 %
829 %
830 %
831 %
832 %
833 %
834 %
835 %
836 %
837 %
838 %
839 %
840 %
841 %
842 %
843 %
844 %
845 %
846 %
847 %
848 %
849 %
850 %
851 %
852 %
853 %
854 %
855 %
856 %
857 %
858 %
859 %
860 %
861 %
862 %
863 %
864 %
865 %
866 %
867 %
868 %
869 %
870 %
871 %
872 %
873 %
874 %
875 %
876 %
877 %
878 %
879 %
880 %
881 %
882 %
883 %
884 %
885 %
886 %
887 %
888 %
889 %
890 %
891 %
892 %
893 %
894 %
895 %
896 %
897 %
898 %
899 %
900 %
901 %
902 %
903 %
904 %
905 %
906 %
907 %
908 %
909 %
910 %
911 %
912 %
913 %
914 %
915 %
916 %
917 %
918 %
919 %
920 %
921 %
922 %
923 %
924 %
925 %
926 %
927 %
928 %
929 %
930 %
931 %
932 %
933 %
934 %
935 %
936 %
937 %
938 %
939 %
940 %
941 %
942 %
943 %
944 %
945 %
946 %
947 %
948 %
949 %
950 %
951 %
952 %
953 %
954 %
955 %
956 %
957 %
958 %
959 %
960 %
961 %
962 %
963 %
964 %
965 %
966 %
967 %
968 %
969 %
970 %
971 %
972 %
973 %
974 %
975 %
976 %
977 %
978 %
979 %
980 %
981 %
982 %
983 %
984 %
985 %
986 %
987 %
988 %
989 %
990 %
991 %
992 %
993 %
994 %
995 %
996 %
997 %
998 %
999 %
1000 %
    
```

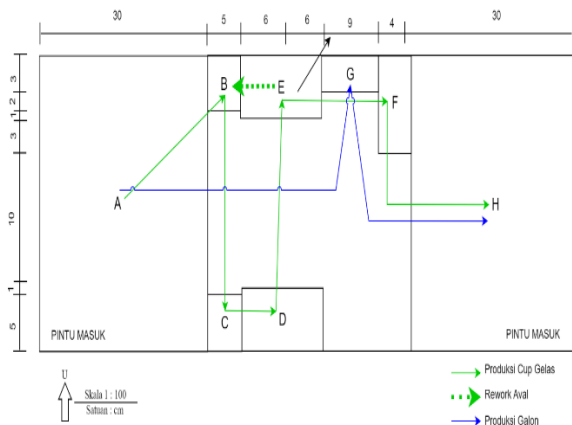
Gambar 3. Coding Squirrel Search Algorithm

Dalam penelitian ini, algoritma SSA digunakan untuk mendapatkan alternatif urutan departemen. Parameter ditetapkan oleh peneliti sebesar 1000 untuk jumlah iterasi dan 100, 300 dan 500 untuk jumlah populasi. Selanjutnya *coding* di-*running* masing-masing sebanyak 3 kali pada setiap kombinasi jumlah populasi dan iterasi. Penetapan nilai populasi menggunakan angka yang relatif besar dengan pertimbangan semakin banyak populasi yang digunakan semakin banyak pula jumlah tupai yang akan diteliti. Selanjutnya jumlah iterasi yang banyak dapat membantu tupai untuk menentukan lokasi yang optimal karena melakukan pembaruan lokasi secara terus menerus hingga iterasi terpenuhi. Pembaruan lokasi tersebut akan menghasilkan urutan departemen yang lebih mendekati optimal. Hasil pengolahan data menggunakan SSA pada MATLAB ditunjukkan pada Tabel 3. Hasil urutan departemen dan total jarak perpindahan material berbeda-beda pada setiap kombinasi jumlah populasi dan iterasi. Hal ini disebabkan ruang pencarian solusi dengan algoritma SSA memiliki nilai acak yang berbeda.

Berdasarkan pengolahan data menggunakan algoritma *modified* SSA, *layout* dengan urutan departemen A-B-E-G-F-C-D-H merupakan solusi terbaik karena menghasilkan total jarak perpindahan material paling minimum sebesar 2731,5 m. Gambar 4 menunjukkan *layout* usulan berdasarkan solusi terbaik yang dihasilkan. Koordinat titik tengah tiap departemen berdasarkan *layout* usulan dapat dilihat pada Tabel 4.

Tabel 3. Alternatif *Layout* Usulan

Populasi	Iterasi	Replikasi	Total Jarak Perpindahan (m)	Urutan Departemen
100	1000	1	3371,5	H-B-C-D-E-F-G-A
		2	3358,5	H-B-F-C-E-G-D-A
		3	3242,5	A-B-C-D-E-F-G-H
300	1000	1	3611,5	H-G-F-D-C-B-E-A
		2	3407,5	A-C-D-F-G-E-B-H
		3	3083,5	A-B-E-F-G-D-C-H
500	1000	1	2738,5	A-B-E-F-G-C-D-H
		3	2731,5	A-B-E-G-F-C-D-H
		3	2783,5	A-B-E-F-G-C-D-H



Gambar 4. *Layout* Usulan

Tabel 4. Koordinat Titik Tengah *Layout* Usulan

No.	Kode	Titik Koordinat	
		X	Y
1.	A	15	12,5
2.	B	32,5	22,5
3.	C	32,5	2,5
4.	D	38	6
5.	E	41	22
6.	F	58	20,5
7.	G	51,5	23,5
8.	H	75	12,5

Tabel 5. Koordinat Titik Tengah Kondisi Saat ini

No.	Kode	X _i	Y _i
1.	A	75	12,5
2.	B	57,5	5,5
3.	C	57,5	22,5
4.	D	49	22
5.	E	45	6
6.	F	53	7,5
7.	G	53,5	1,5
8.	H	15	12,5

Sebelum dilakukan perbandingan total jarak perpindahan material antara *layout* perusahaan pada kondisi saat ini dengan usulan maka harus menentukan

titik tengah dari setiap departemen pada *layout* (Gambar 2). Koordinat titik tengah dari setiap departemen disajikan pada Tabel 5. Setelah titik tengah setiap departemen ditentukan, jarak antar departemen dihitung dengan menggunakan metode *rectilinear*. Selanjutnya total jarak perpindahan material pada *layout* perusahaan dihitung dengan menggunakan persamaan (7). Hasil total perpindahan material ditunjukkan pada Tabel 6. *Layout* yang digunakan perusahaan saat ini memiliki total perpindahan material sebesar 3103,5 meter.

Tabel 6. Total Jarak Perpindahan Material

Dept Asal (i)	Dept. Tujuan (j)	d _{ij}	F	D _{ij}
A	C	27,5	8	220
A	G	30,5	4	122
A	F	27	1	27
B	C	17	12	204
B	E	13	21	273
C	D	9	18	162
D	E	20	21	420
E	F	9,5	20	190
F	H	43	24	1032
G	H	49,5	9	445,5
Total				3103,5

Perbandingan total jarak perpindahan material antara *layout* perusahaan dengan *layout* usulan dengan total jarak perpindahan material berkurang sebesar 372 meter (Tabel 7). Tingkat efisiensi tata letak usulan dengan algoritma *modified* SSA sebesar 11,99%. Pengurangan jarak disebabkan adanya perpindahan letak departemen. Dua departemen yang memiliki frekuensi perpindahan material yang tinggi yaitu departemen *thermoforming* (E) ke departemen *crusher* (B) dan departemen *packing* (F) ke gudang barang jadi (H) lebih didekatkan pada *layout* usulan. Selain itu, aliran *backtracking* yang terjadi dari departemen F ke H sudah dieliminasi dengan perpindahan lokasi tersebut.

Tabel 2. Perbandingan Performansi *Layout*

Total Jarak Perpindahan Material			
Perusahaan (m)	Usulan (m)	Selisih (m)	Efisiensi
3103,5	2731,5	372	11,99%

Pada penelitian terdahulu yang dilakukan oleh Al-Saati & Algarni (2022), algoritma *modified* SSA terbukti

menghasilkan solusi yang lebih baik untuk menyelesaikan *single-row facility layout problem*. Hasil penelitian ini membuktikan bahwa algoritma *modified SSA* juga menunjukkan performansi bagus dalam menyelesaikan *two-row facility layout problem*. Hasil penelitian ini sejalan dengan hasil penelitian Faiz *et al.*, (2022) dan Daya *et al.*, (2018) dimana perancangan ulang tata letak fasilitas mampu menurunkan total biaya atau jarak perpindahan material dibanding dengan layout perusahaan saat ini. Igoritma *modified SSA* yang diusulkan agar mampu memberikan efisiensi yang lebih besar dari 11,99% maka perlu dilakukan parameter tuning dengan menggunakan desain eksperimen dan penambahan jumlah replikasi.

4. KESIMPULAN

Hasil perancangan ulang tata letak fasilitas menggunakan *modified Squirrel Search Algorithm* (SSA) menunjukkan tata letak fasilitas usulan lebih efisien dibandingkan tata letak fasilitas perusahaan saat ini. Perpindahan letak departemen menyebabkan berkurangnya total jarak perpindahan material sebesar 372 m atau sebesar 11,99%. Bagi penelitian selanjutnya, performansi *modified SSA* perlu dibandingkan dengan algoritma *hybrid SSA* atau algoritma metaheuristik lainnya.

DAFTAR PUSTAKA

- Al-Saati, N. A., & Algamal, Z. Y. (2022). A modified squirrel search algorithm for solving facility layout problems. *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, 20(2), 348. <https://doi.org/10.12928/telkonnika.v20i2.21845>
- Cravo, G. L., & Amaral, A. R. S. (2019). A GRASP algorithm for solving large-scale single row facility layout problems. *Computers & Operations Research*, 106, 49–61. <https://doi.org/10.1016/j.cor.2019.02.009>
- Daya, M. A., Sitania, F. D., & Profita, A. (2019). Perancangan Ulang (re-layout) tata letak fasilitas produksi dengan metode blocplan (studi kasus: ukm roti rizki, Bontang). *PERFORMA Media Ilmiah Teknik Industri*, 17(2), 140–145. <https://doi.org/10.20961/performa.17.2.29664>
- Elkassas, A. M., & ElWakil, M. (2019). Facility Layout Problem Using Salp Swarm Algorithm. *2019 6th International Conference on Control, Decision and Information Technologies (CoDIT)*, 1859–1864. <https://doi.org/10.1109/CoDIT.2019.8820422>
- Faiz, N. M., Sugiyono, A., & Bernadhi, B. D. (2022). Usulan Perbaikan Tata Letak Fasilitas PT. Promanufacture Indonesia Menggunakan Aplikasi Blocplan. *Prosiding Konstelasi Ilmiah Mahasiswa UNISSULA*, 210–222. <https://jurnal.unissula.ac.id/index.php/kimueng/article/view/20569>
- Garcia-Hernandez, L., Salas-Morera, L., Carmona-Munoz, C., Abraham, A., & Salcedo-Sanz, S. (2020). A Hybrid Coral Reefs Optimization—Variable Neighborhood Search Approach for the Unequal Area Facility Layout Problem. *IEEE Access*, 8(1), 134042–134050. <https://doi.org/10.1109/ACCESS.2020.3010577>
- Guan, C., Zhang, Z., Liu, S., & Gong, J. (2019). Multi-objective particle swarm optimization for multi-workshop facility layout problem. *Journal of Manufacturing Systems*, 53(September), 32–48. <https://doi.org/10.1016/j.jmsy.2019.09.004>
- Kaveh, A., & Vazirinia, Y. (2019). Construction Site Layout Planning Problem Using Metaheuristic Algorithms: A Comparative Study. *Iranian Journal of Science and Technology, Transactions of Civil Engineering*, 43(2), 105–115. <https://doi.org/10.1007/s40996-018-0148-6>
- Krömer, P., Platoš, J., & Snášel, V. (2020). Solving the single row facility layout problem by differential evolution. *Proceedings of the 2020 Genetic and Evolutionary Computation Conference*, 210–218. <https://doi.org/10.1145/3377930.3389839>
- Kusumaningsih, D. A., Azim, A. F., Albab, D. A. I. U., Hans, F. R., Korin, F., Pohan, R. N. A. M., Ananta, V. S., & Rifai, A. P. (2022). Simulated Annealing untuk Perancangan Tata Letak Industri Furniture dengan Model Single dan Double Row Layout. *Jurnal Media Teknik Dan Sistem Industri*, 6(1), 60–67. <https://doi.org/10.35194/jmtsi.v6i1.1773>
- Liu, J., & Liu, J. (2019). Applying multi-objective ant colony optimization algorithm for solving the unequal area facility layout problems. *Applied Soft Computing*, 74, 167–189. <https://doi.org/10.1016/j.asoc.2018.10.012>
- Muharni, Y., Irman S M, A., & Noviansyah, Y. (2020). Perancangan Tata Letak Gudang Barang Jadi Menggunakan Kebijakan Class-Based Storage dan Particle Swarm Optimization Di PT XYZ. *JURNAL TEKNIK INDUSTRI*, 10(3), 200–209. <https://doi.org/10.25105/jti.v10i3.8405>
- Prayogo, D., Eric, S., Sutanto, J. C., & Suryo, H. E. (2018). Optimasi Tata Letak Fasilitas Proyek Konstruksi dengan Menggunakan Algoritma Metaheuristik. *Prosiding Seminar Nasional Ilmu Terapan*, 1(1), 1–8. <https://repository.petra.ac.id/20304/>
- Rajak, S. (2018). Optimasi Tata Letak Fasilitas Produksi Menggunakan Algoritma Genetika. Thesis. Universitas Islam Indonesia. <https://dspace.uui.ac.id/bitstream/handle/123456789/6570/Tesis%20pdf.pdf?sequence=1>
- Senatra, A. F. E., & Widyadana, I. G. A. (2022). Perbaikan Tata Letak Fasilitas dengan Pendekatan Simulasi dan Evolutionary Algorithm pada Segmen I PT X. *Jurnal Titra*, 10(1), 1–8. <https://publication.petra.ac.id/index.php/teknik-industri/article/view/9873>
- Singh, D., & Ingole, S. (2019). Multi-objective facility layout problems using BBO, NSBBO and NSGA-II metaheuristic algorithms. *International Journal of Industrial Engineering Computations*, 10(2), 239–262. <https://doi.org/10.5267/j.ijiec.2018.6.006>
- Sun, X., Chou, P., Koong, C.-S., Wu, C.-C., & Chen, L.-R. (2022). Optimizing 2-opt-based heuristics on GPU for solving the single-row facility layout problem. *Future Generation Computer Systems*,

- 126, 91–109.
<https://doi.org/10.1016/j.future.2021.07.022>
- Tang, H., Ren, S., Jiang, W., & Chen, Q. (2022). Optimization of Multi-Objective Unequal Area Facility Layout. *IEEE Access*, 10(April), 38870–38884.
<https://doi.org/10.1109/ACCESS.2022.3163287>
- Tongur, V., Hacibeyoglu, M., & Ulker, E. (2020). Solving a big-scaled hospital facility layout problem with meta-heuristics algorithms. *Engineering Science and Technology, an International Journal*, 23(4), 951–959.
<https://doi.org/10.1016/j.jestch.2019.10.006>
- Wang, Z., & Xu, C. (2023). A thin-provisioned and functionalized memetic algorithm for the single row facility layout problem. *Applied Soft Computing*, 146, 110716.
<https://doi.org/10.1016/j.asoc.2023.110716>
- Wignjosuebrotto, S. (2009). *Tata Letak Pabrik dan Pемindahan Bahan* (3rd ed.). Surabaya: Guna Widya.
<https://onesearch.id/Record/IOS18055.slims-3350>