Documentație completă 2SGA – FJSP

# Structura generală a algoritmului 2SGA

Algoritmul 2SGA (Two-Stage Genetic Algorithm) este un algoritm metaheuristic dezvoltat pentru problema Flexible Job Shop Scheduling Problem (FJSP).  
El integrează două etape majore:  
- Etapa 1: generarea unei ordini de operații (OS) și alocarea mașinilor pe baza unei selecții greedy (cea mai rapidă mașină).  
- Etapa 2: codificare OS+MA fixă, aplicarea operatorilor genetici (crossover, mutation), evaluare și selecție a celei mai bune soluții.

# 1. Joburi și Constrângeri (extrase din SetDeDate\_Lucrare1.fjs)

Sunt 8 joburi în total, fiecare cu un număr de operații și mașini eligibile per operație.

J1 – 2 operații

J2 – 2 operații

J3 – 2 operații

J4 – 2 operații

J5 – 2 operații

J6 – 2 operații

J7 – 1 operații

J8 – 1 operații

## Constrângeri de precedență între operații:

J3O2 ← J1O1, J1O2

J4O1 ← J2O1, J2O2

J4O2 ← J3O1, J3O2, J4O1

**Rezolvare 2SGA – Etapa 1 explicată pas cu pas**

PAS 1: J5 - O1

J5-O1 are 2 mașini eligibile: M1 (timp=14), M3 (timp=10)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=60 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=0 → start=48, end=58 → Case 1

✅ Alegem M3 → timp minim de finalizare. Programare: start=48, end=58, durată=10, Case 1

PAS 2: J7 - O1

J7-O1 are 1 mașini eligibile: M1 (timp=6)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=52 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=52, durată=6, Case 1

PAS 3: J4 - O2

J4-O2 are 2 mașini eligibile: M2 (timp=6), M3 (timp=6)

Predecesori: J3O1, J3O2, J4O1

> M2: last\_machine\_time=46, pred\_end=48, prev\_op\_end=0 → start=48, end=54 → Case 2

> M3: last\_machine\_time=48, pred\_end=48, prev\_op\_end=0 → start=48, end=54 → Case 1

✅ Alegem M2 → timp minim de finalizare. Programare: start=48, end=54, durată=6, Case 2

PAS 4: J4 - O1

J4-O1 are 1 mașini eligibile: M2 (timp=24)

Predecesori: J2O1, J2O2

> M2: last\_machine\_time=46, pred\_end=35, prev\_op\_end=0 → start=46, end=70 → Case 1

✅ Alegem M2 → timp minim de finalizare. Programare: start=46, end=70, durată=24, Case 1

PAS 5: J8 - O1

J8-O1 are 1 mașini eligibile: M3 (timp=8)

Predecesori: niciunul

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=0 → start=48, end=56 → Case 1

✅ Alegem M3 → timp minim de finalizare. Programare: start=48, end=56, durată=8, Case 1

PAS 6: J6 - O2

J6-O2 are 2 mașini eligibile: M2 (timp=7), M3 (timp=9)

Predecesori: niciunul

> M2: last\_machine\_time=46, pred\_end=0, prev\_op\_end=46 → start=46, end=53 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=46 → start=48, end=57 → Case 1

✅ Alegem M2 → timp minim de finalizare. Programare: start=46, end=53, durată=7, Case 1

PAS 7: J3 - O2

J3-O2 are 1 mașini eligibile: M3 (timp=15)

Predecesori: J1O1, J1O2

> M3: last\_machine\_time=48, pred\_end=46, prev\_op\_end=48 → start=48, end=63 → Case 1

✅ Alegem M3 → timp minim de finalizare. Programare: start=48, end=63, durată=15, Case 1

PAS 8: J2 - O1

J2-O1 are 2 mașini eligibile: M2 (timp=12), M3 (timp=10)

Predecesori: niciunul

> M2: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=58 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=0 → start=48, end=58 → Case 1

✅ Alegem M2 → timp minim de finalizare. Programare: start=46, end=58, durată=12, Case 1

PAS 9: J5 - O2

J5-O2 are 2 mașini eligibile: M2 (timp=15), M3 (timp=10)

Predecesori: niciunul

> M2: last\_machine\_time=46, pred\_end=0, prev\_op\_end=10 → start=46, end=61 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=10 → start=48, end=58 → Case 1

✅ Alegem M3 → timp minim de finalizare. Programare: start=48, end=58, durată=10, Case 1

PAS 10: J2 - O2

J2-O2 are 1 mașini eligibile: M1 (timp=16)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=19 → start=46, end=62 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=62, durată=16, Case 1

PAS 11: J3 - O1

J3-O1 are 2 mașini eligibile: M1 (timp=17), M3 (timp=20)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=63 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=0 → start=48, end=68 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=63, durată=17, Case 1

PAS 12: J1 - O1

J1-O1 are 2 mașini eligibile: M1 (timp=15), M2 (timp=20)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=61 → Case 1

> M2: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=66 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=61, durată=15, Case 1

PAS 13: J1 - O2

J1-O2 are 2 mașini eligibile: M1 (timp=7), M3 (timp=9)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=39 → start=46, end=53 → Case 1

> M3: last\_machine\_time=48, pred\_end=0, prev\_op\_end=39 → start=48, end=57 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=53, durată=7, Case 1

PAS 14: J6 - O1

J6-O1 are 2 mașini eligibile: M1 (timp=5), M2 (timp=7)

Predecesori: niciunul

> M1: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=51 → Case 1

> M2: last\_machine\_time=46, pred\_end=0, prev\_op\_end=0 → start=46, end=53 → Case 1

✅ Alegem M1 → timp minim de finalizare. Programare: start=46, end=51, durată=5, Case 1

Etapa 2 – Codificare OS + MA și operatori genetici (stil foaie și pix)

Am selectat două poziții valide pentru mutație (swap):

Pozițiile 1 ↔ 2 → swap între 9 și 13

Noua secvență OS (după OSM): [13, 9, 12, 3, 14, 10, 4, 1, 5, 2, 11]

MA rămas neschimbat: [3, 1, 2, 2, 3, 3, 1, 2, 3, 1, 2]

Decodare pas cu pas a OS+MA mutate

PAS 1: J7 - O1 pe M3 → start=0, end=999 | predec\_end=0, prev\_op\_end=0, mach\_end=0

PAS 2: J5 - O1 pe M1 → start=0, end=14 | predec\_end=0, prev\_op\_end=0, mach\_end=0

PAS 3: J6 - O2 pe M2 → start=0, end=7 | predec\_end=0, prev\_op\_end=0, mach\_end=0

PAS 4: J2 - O1 pe M2 → start=7, end=19 | predec\_end=0, prev\_op\_end=0, mach\_end=7

PAS 5: J8 - O1 pe M3 → start=999, end=1007 | predec\_end=0, prev\_op\_end=0, mach\_end=999

PAS 6: J5 - O2 pe M3 → start=1007, end=1017 | predec\_end=0, prev\_op\_end=14, mach\_end=1007

PAS 7: J2 - O2 pe M1 → start=19, end=35 | predec\_end=0, prev\_op\_end=19, mach\_end=14

PAS 8: J1 - O1 pe M2 → start=19, end=39 | predec\_end=0, prev\_op\_end=0, mach\_end=19

PAS 9: J3 - O1 pe M3 → start=1017, end=1037 | predec\_end=0, prev\_op\_end=0, mach\_end=1017

PAS 10: J1 - O2 pe M1 → start=39, end=46 | predec\_end=0, prev\_op\_end=39, mach\_end=35

PAS 11: J6 - O1 pe M2 → start=39, end=46 | predec\_end=0, prev\_op\_end=0, mach\_end=39

✅ Makespan final după decodare: 1037 unități de timp.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Pas | Job | Operație | Mașină aleasă | Start | End |  |
| 1 | J5 | O1 | 3 | 48 | 58 |  |
| 2 | J7 | O1 | 1 | 46 | 52 |  |
| 3 | J4 | O2 | 2 | 48 | 54 |  |
| 4 | J4 | O1 | 2 | 46 | 70 |  |
| 5 | J8 | O1 | 3 | 48 | 56 |  |
| 6 | J6 | O2 | 2 | 46 | 53 |  |
| 7 | J3 | O2 | 3 | 48 | 63 |  |
| 8 | J2 | O1 | 2 | 46 | 58 |  |
| 9 | J5 | O2 | 3 | 48 | 58 |  |
| 10 | J2 | O2 | 1 | 46 | 62 |  |
| 11 | J3 | O1 | 1 | 46 | 63 |  |
| 12 | J1 | O1 | 1 | 46 | 61 |  |
| 13 | J1 | O2 | 1 | 46 | 53 |  |
| 14 | J6 | O1 | 1 | 46 | 51 |  |

Makespan final (Etapa 1): 70 unități de timp.