Ant colony system for scheduling problem

Ambrosini, Luca Nicolò, Giancarlo

[1]Universitat Politècnica de València

May, 8th 2017

Overview

Table of Contents

Idea

Aim

Novel work to tackle literature leaks in the scheduling domain applying meteheuristic approaches.

Scheduling domain:

- Energy aware scheduling
- Realistic problem (setup, release, due date)

Metaheuristic:

- Ant Colony Optimization
- Ant Colony System

Motivation

Why scheduling problem?

- most of daily life problem can easily map in scheduling problem
    - industrial/manufacturing environment
    - supply-chain management
    - space orbit
- optimization of shared resources
- search for efficiency and cost reduction
- avoid critical situation

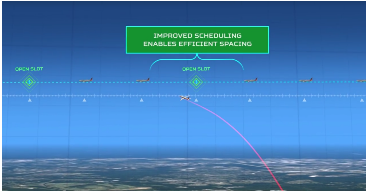

Table of Contents

Problem description

The problem consists in *scheduling a set of job*, on a set of *unrelated parallel machines*, while optimizing objective functions.

Job characteristics are:

- *release date* (hard constraint)
- *due date* (soft constraint)
- *weighted penalty cost*

Each job have to be processed by only one machine, selected from a set of eligible ones, where *processing time* and *energy consumption* depend by the selected machine.

In order to process a job the machines must be setup and this setup time depends on the *machine* and the *previously scheduled job* on it.

$$R_m | M_j, p_{ij}, E_{ij}, r_j, d_j, w_j, s_{ijk} | \sum w_j T_j, \sum E_{ij}, \sum S_{ijk}$$

Problem example

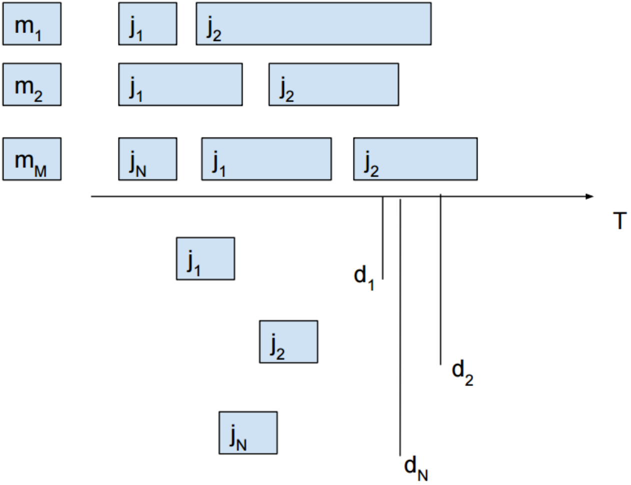

Figure: Scheduling example 1

Problem example

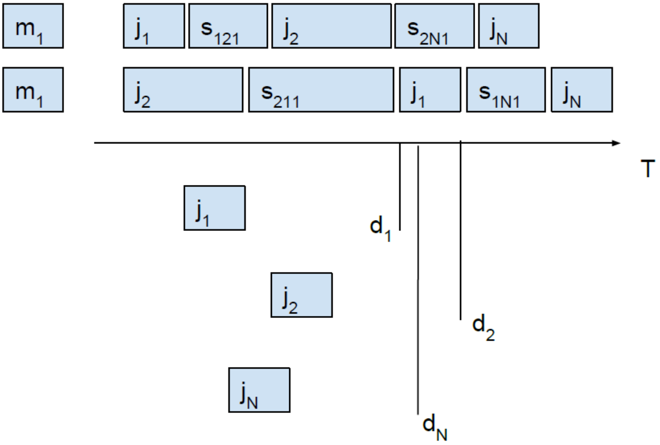

Figure: Scheduling example 2

Problem objectives

In relation to a possible scheduling solution s, three different objective functions can be stated:

- $TT(s)$: total weighted tardiness of the jobs,
- $EN(s)$: energy consumption,
- $ST(s)$: total setup time.

The aim is to simultaneously optimized them in a multi-objective function:

$$s^* = \arg\min_{s \in S}[TT(s), EN(s), ST(s)] \tag{1}$$

where S denotes the feasibility space for the problem solutions.

Problem objectives

The three objective functions in (1) are aggregated into a scalar normalized objective function F:

$$F(s) = \sum_{g=1}^{3} \Pi_g \cdot \frac{f_g(s) - f_g^-}{f_g^+ - f_g^-} \tag{2}$$

where:

- $f_g(s)$, $g \in \{1, 2, 3\}$, represents $TT(s)$, $EN(s)$ and $ST(s)$.
- f_g^- represents the best (*minimum*) value
- f_g^+ is an estimation of the worse value for $f_g(s)$
- Π_g, $g \in \{1, 2, 3\}$, represents the relative importance given by the decision maker to the different objective, where $\sum_g \Pi_g = 1$

Table of Contents

Ant colony

- Meta-heuristic proposed by [Colorni, et Al], improved by [Dorigo, Gambardella, et al.]
- Mimicry of ants behavior to find optimal path between nest and food
    - Transition rules
- Indirect communication between ants through pheromones amount
    - Update rules
- State of the art in various optimization problem

Ant exaple

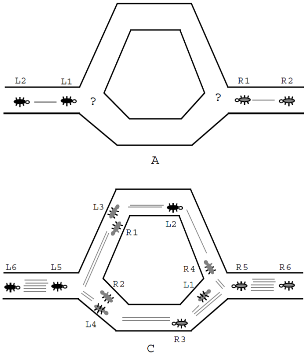

Figure: How real ants find a shortest path [Dorigo, Gambardella, et al.].

Transition rules

Ant Colony Optimization

$$P_{ij}^{k} = \frac{\tau_{ij}^{\alpha}\ \eta_{ij}^{\beta}}{\sum_{l \in \Psi} \tau_{il}^{\alpha}\ \eta_{il}^{\beta}}$$

Ant Colony System

Pseudo random proportional rule:

$$j = \begin{cases} \underset{j \in J(i)}{\arg\max}\{[\tau_{ij}] \cdot [\eta_{ij}]^{\beta}\} & \text{if } q \leq q_0 \quad \text{(exploitation)} \\ S & \textit{otherwise} \quad \text{(biased exploration)} \end{cases}$$

where S is a random variable selected according to P_{ij}^{k}

Update rules

Global update rule

$$\tau_{ij} \leftarrow (1-\alpha)\cdot\tau_{ij} \; + \; \alpha\cdot\tau_{ij}^{k}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} 1/L_{gb} & \text{if arc } (i,j) \in \text{ global-best-tour} \\ 0 & \text{Otherwise} \end{cases}$$

Local update rule

$$\tau_{ij} \leftarrow (1-\rho)\cdot\tau_{ij} \; + \; \rho\cdot\tau_{ij}^{k}$$

$$\Delta\tau_{ij}^{k} = \begin{cases} 1/L_{init} & \text{if arc } (i,j) \in \text{ initial-tour} \\ 0 & \text{Otherwise} \end{cases}$$

Solution encoding

Machine assignment

$$S_1 = [\ 3\ 2\ 3\ 1\ 4\ 3\ 4\ 2\ 1\ 4] \qquad \text{(stage 1)}$$

(interpretation) machine three (m_3) has assigned jobs 1, 3 and 6

Jobs sequencing

$$S_2 = \begin{bmatrix} 9 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 8 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 6 & 3 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 5 & 10 & 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad \text{(stage 2)}$$

(interpretation) machine one (m_1) will process jobs in the following order: $9 \rightarrow 4$

Transition definition (Stage 1)

Transition rule

$$j = \begin{cases} \arg\max\limits_{j \in J(i)} \{[\tau^I_{jk}] \cdot [\eta^I_{jk}]^\beta\} & \text{if } q \leq q_0 \quad \text{(exploitation)} \\ S & \textit{otherwise} \quad \text{(biased exploration)} \end{cases} \tag{3}$$

Visibility metric

$$\eta^I_{jk} = \frac{1}{P_{jk}} \tag{4}$$

$$\eta^I_{jk} = \frac{1}{\left[\frac{P_{jk}}{\max_{m \in M_j}(P_{jm})} + \frac{E_{jk}}{\max_{m \in M_j}(E_{jm})}\right]} \tag{5}$$

[Tonelli, Salido, et al. 2016]

Transition definition (Stage 2)

Transition rule

$$j = \begin{cases} \arg\max\limits_{j \in J(i)} \{[\tau_{ij}^{II,k}] \cdot [\eta_{ij}^{II,k}]^{\beta}\} & \text{if } q \leq q_0 \quad \text{(exploitation)} \\ S & \textit{otherwise} \quad \text{(biased exploration)} \end{cases} \tag{6}$$

Visibility metric

$$\eta_{ij}^{II,k} = \frac{1}{s_{ijk}} \tag{7}$$

$$\eta_{ij}^{II,k} = \frac{1}{\left[\frac{s_{ijk}}{\max_{j' \in J_k}(s_{ij'k})} + \frac{r_j}{\max_{j' \in J_k}(r_{j'})}\right]} \tag{8}$$

Pheromone global update

Stage 1

$$\tau^{I}_{jk} \leftarrow (1-\alpha)\cdot\tau^{I}_{jk} + \alpha\cdot\Delta\tau^{I,Best}_{jk} \tag{9}$$

$$\Delta\tau^{I,Best}_{jk} = \begin{cases} 1/F(s_{gb}) & \text{if arc } (j,k) \in \text{ global-best-schedule} \\ 0 & \text{Otherwise} \end{cases} \tag{10}$$

Stage 2

$$\tau^{II,k}_{ij} \leftarrow (1-\alpha)\cdot\tau^{II,k}_{ij} + \alpha\cdot\Delta\tau^{II,Best}_{ij} \tag{11}$$

$$\Delta\tau^{II,Best}_{ij} = \begin{cases} 1/F(s_{gb}) & \text{if arc } (i,j,k) \in \text{ global-best-schedule} \\ 0 & \text{Otherwise} \end{cases} \tag{12}$$

Pheromone local update

Stage 1

$$\tau^{I}_{jk} \leftarrow (1-\rho)\cdot\tau^{I}_{jk} + \rho\cdot\Delta\tau^{I,Best}_{jk} \tag{13}$$

$$\Delta\tau^{I,Best}_{jk} = \begin{cases} 1/F(s_{init}) & \text{if arc } (j,k) \in \text{ initial-best-schedule} \\ 0 & \text{Otherwise} \end{cases} \tag{14}$$

Stage 2

$$\tau^{II,k}_{ij} \leftarrow (1-\rho)\cdot\tau^{II,k}_{ij} + \rho\cdot\Delta\tau^{II,Best}_{ij} \tag{15}$$

$$\Delta\tau^{II,Best}_{ij} = \begin{cases} 1/F(s_{init}) & \text{if arc } (i,j,k) \in \text{ initial-best-schedule} \\ 0 & \text{Otherwise} \end{cases} \tag{16}$$

Local search procedure

Algorithm 1: Local search procedure

Set $LocalIteration = 1$;
while $LocalIteration \leq MaxLocalIterations$ **do**
    Generate random variable (rv) from $U(0, 1)$;
    if $rv < 0.5$ **then**
        generate neighboring solution for $S_1(Ant)$: $N_1(Ant) \wedge x = 1$;
    else
        generate neighboring solution for $S_2(Ant)$: $N_2(Ant) \wedge x = 2$
    end
    Determine $F(N_x(Ant))$;
    if $F(N_x(Ant)) \leq F(Ant)$ **then**
        $S_x \leftarrow N_x(Ant)$;
    end
    $LocalIteration = LocalIteration + 1$
end

Ant colony system workflow

Algorithm 2: ACS workflow

Populate the paths with specified pheromone amounts $(\tau_{jk}^{I}, \tau_{ij}^{II,k})$;
while *not StopCriteria* **do**
    Set $Step = 1$;
    while $Step \leq MaxSteps$ **do**
        for $Ant \in Ants$ **do**
            Solve *Step* for Stage 1 (Assignment): find S_1 according to eqs. (3) to (5);
            Solve *Step* for Stage 2 (Sequencing): find S_2 according to eqs. (6) to (8);
        end
        Update pheromone amounts locally according to eqs. (13) and (15);
    end
    Calculate $F(Ants)$ that are associated with S_1 and S_2;
    Execute local search procedure for all ants;
    Update pheromone amounts globally according to eqs. (9) and (11);
end

Table of Contents

Experimental comparison

The problem instances are grouped in sets over the number of jobs n and involved machines m, where possible set dimension are:

Jobs	Machines
30	4
50	6
100	10
250	20

- each different sets contains 125 instances
- all solving strategies run on a 2.4 GHz Intel Core 2 Duo
- objectives weight are fixed to $\Pi_1 = 0.6$, $\Pi_2 = 0.35$ and $\Pi_3 = 0.05$

Experimental comparison

Instances		No-L.S.		L.S.	
n	m	Baseline	eqs. (5) and (8)	Baseline	eqs. (5) and (8)
30	4	0,13297	0,13161	0,13033	0,13221
50	6	0,12810	0,11930	0,12210	0,12033
100	10	0,11653	0,11460	0,10653	0,10260
250	20	0,05187	0,04876	0,04187	0,04076

Table: Average multi-objective values reached in the different classes of instances. Stop criteria set to 600 seconds.

Table of Contents

Conclusion

- Modelling of the analyzed problem as ACO and ACS
- Defined visibility metrics that better capture problem features
- Adapted a well know ant colonony local search procedure
- Empirically evaluated the different settings of the algorithm
- Useful insight over the algorithm settings