A GRASPxELS for supply chain optimization considering payment delay between members

S. Kemmoe Tchomte[1], D. Lamy[2] and N. Tchernev[2]

1. CRCGM EA 38 49, 28 place Henri Dunant, 63000 Clermont-Ferrand
Syverin.KEMMOE_TCHOMTE@udamail.fr

2. LIMOS UMR 6158, Complexe scientifique des Cézeaux, 63173 AUBIERE cedex - FRANCE
lamy@poste.isima.fr, nikolay.tchernev@udamail.fr

Keywords: Supply *chain, payment delay, GRASP, job shop.*

1 Introduction

This paper addresses the supply chain optimization considering financial exchange between different plants and different partners taking into account the delays of payments. The financial exchange is considered as the cash flow exchanges depending on the payment terms between all the partners of the supply chain. We study the flow of money in a supply chain (SC) where each supply chain partner receives money from the downstream partners and makes payments to the upstream partners. The target is to obtain trade-off solutions during the operational supply chain planning avoiding day-to-day negative cash flow.

Supply chain can be defined as a system of logistic units (LU) which can be suppliers, manufacturers, distributors, retailers and customers where material, financial and information flows connect participants in both directions. At each logistic unit a logistic activity is executed. Figure 2summarizes and presents a logistic activity i executed by LU_j. This LU_j can be any one of SC members. According to the figure, a logistic activity has a duration p_i time units. The first supplier will receive account payable AP_{i_1} (Financial resources) from the logistic unit (cash outflow) after a delay of α_i time units and the second supplier will receive after a delay of β_i times AP_{i2}. After the end of the logistic activity i, the SC LU_j will receive account receivable AR_i (financial resources) according to a delay of δ_i time units (cash inflow).Each logistic activity has duration which can be composed of different delays and therefore there are three sub-activities; each sub-activity has the each owns duration, cost and account (receivable or payable).

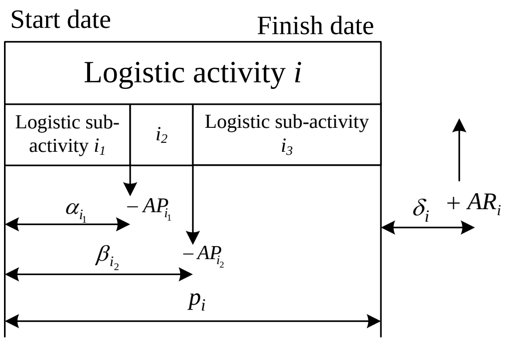

Fig. 1: Logistic activity description

To solve the problem we chose the modelling approach proposed by [1] which approach is based on Job Shop (JS) theoretical model where each machine represents a supply chain member and jobs represent the product batches. Each member of the supply chain has its own cash amount available and uses it to pay the upstream partners (suppliers and/or plant). The Job-Shop Scheduling Problem (JSSP) is a well-known optimization problem often used in practical scheduling applications in the manufacturing sector (The *JSSP* can be formulated as follows: a set of n jobs $(i = 1, 2, \ldots n)$ has to be processed on a set of m machines $(j = 1, 2, \ldots m)$. Each job is fully defined by an ordered (linear) sequence of operations that are associated with a particular machine. In addition, the JSSP must satisfy other constraints such as: *(i)* no more than one operation of any job can be executed simultaneously; *(ii)* no machine can process more than one operation at the same time; *(iii)* the job operations must be executed in a predefined sequence and once an operation is started, no pre-emption is permitted. In addition to classical *JSSP* there are two extra constraints: *(iv)* financial each operation can start if there is enough financial resource to pay upstream partners; *(v)* no-wait between some operations which model logistic sub-activities (see Fig. 1 sub-activities i_1, i_2 and i_3). The objective is to minimize the finish dates of all activities taking into account the *JS*, no-wait and financial constraints. Recently [1] proposed a linear formulation which solves small instances. In this paper we propose a metaheuristic approach to solve large scale instances.

2 Framework definition

The framework consists in solving a *JSSP* with no-wait and financial constraints using a *GRASPxELS* which is a metaheuristic approach base on Greedy Randomized Adaptive Search Procedure (*GRASP*) hybridized with an evolutionary local search (*ELS*). *GRASP* is a multi-start local search metaheuristic in which each initial solution is constructed using a greedy randomized heuristic. The multi-start approach of the *GRASP* provides *np* > 1 initial solutions, improved by a local search.Using mutation and local search at each iteration *ELS* generates *ns* > 1 children-solutions and selects the best one.

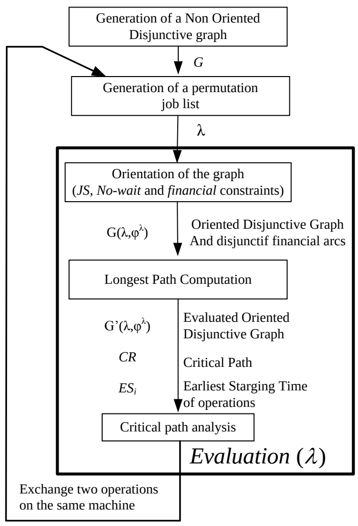

Fig. 1. Optimization framework

The *GRASPxELS* uses a Bierwith encoding [2] which permits to obtain acyclic graphs. Figure 2 shows the optimization framework which takes advantages of the powerful scheme introduced in 1992 by [3]. The JSSP has received a large amount of attention, focusing on meta-heuristics for large instances (see *GA* [4] recently). This master/slave resolution framework is based on a*GRASPxELS* (for job sequence generation on machines) coupled with a powerful local search procedure. The problem is first modeled through simultaneous use of a non-oriented disjunctive graph *G* and flow network G^{FN} to take into account financial constraints. A constructive heuristic and Bierwirth' sequence λ are used then to obtain both an oriented disjunctive graph $G(\lambda)$ and a "slave" flow on the network G^{FN} taking into account no-wait constraints. The fully oriented graph $G(\lambda,\varphi^{\lambda})$encompasses both job-shop, no-wait and financial constraints. Earliest starting time of operations (in $G(\lambda,\varphi^{\lambda})$) can be achieved using any Bellman' like longest path algorithm. At the master level, Bierwith' sequence generation λ is devoted to a local search or *ELS* depending on the place in *GRASPxELS* algorithm.

3 Concluding remarks

This study delineates the relationship between supply chain physical and cash flows throughout the supply chain. The problem is modeled as aJSSP with no-wait and financial constraints i.e. scheduling the supply chain activities efficiently while respecting the financial supply chain capacity. The numerical experiment proves that our framework can obtain promising solutions in a rather short computational time. This work is a step forward definition of efficient models for job-shop's like scheduling problems with extra no-wait and cash-flow constraints.

References

[1] S. Kemmoé, P. Lacomme, N. Tchernev (2012). Application of job-shop model for supply chain optimization considering payment delay between members, ILS 2012, August 26-29, Quebec, Canada.

[2] C. Bierwirth (1995). A generalized permutation approach to jobshop scheduling with genetic algorithms. *OR Spektrum*, 17, 87-92.

[3] P.J.M.Van Laarhoven, E.H.L.Aarts, J.K. Lenstra (1992).Jobshop scheduling by simulated annealing, *OR*, 40(1), 113-125.

[4] Ren Qing-dao-er-ji, Yuping Wang (2012). A new hybrid genetic algorithm for job shop scheduling problem, *C&OR*, 9(10), 2294-2299.