CIRRELT

Centre interuniversitaire de recherche
sur les réseaux d'entreprise, la logistique et le transport

Interuniversity Research Centre
on Enterprise Networks, Logistics and Transportation

Time-Cost Trade-Offs in Resource-Constraint Project Scheduling Problems with Overlapping Modes

François Berthaut
Robert Pellerin
Nathalie Perrier
Adnène Hajji

February 2011

CIRRELT-2011-10

Bureaux de Montréal :
Université de Montréal
C.P. 6128, succ. Centre-ville
Montréal (Québec)
Canada H3C 3J7
Téléphone : 514 343-7575
Télécopie : 514 343-7121

Bureaux de Québec :
Université Laval
2325, de la Terrasse, bureau 2642
Québec (Québec)
Canada G1V 0A6
Téléphone : 418 656-2073
Télécopie : 418 656-2624

www.cirrelt.ca

Time-Cost Trade-Offs in Resource-Constraint Project Scheduling Problems with Overlapping Modes

François Berthaut[1,2,*], Robert Pellerin[1,2], Nathalie Perrier[1,2], Adnène Hajji[1,3]

[1] Interuniversity Research Centre on Enterprise Networks, Logistics and Transportation (CIRRELT)
[2] Department of Mathematics and Industrial Engineering, École Polytechnique de Montréal, P.O. Box 6079, Station Centre-ville, Montréal, Canada H3C 3A7
[3] Department of Operations and Decision Systems, Université Laval, 2325, de la Terrasse, Québec, Canada G1V 0A6

Abstract. In companies, overlapping is commonly regarded as a promising strategy to accelerate project execution. Overlapping consists in executing in parallel two sequential activities by allowing a downstream activity to start before the end of an upstream activity based on preliminary information. However, overlapping entails rework in downstream activity caused by alteration of information exchanged until finalized information is available and additional coordination and communication. Rework and coordination/communication require additional resources and costs. We investigate the time-cost tradeoffs in resource-constrained project scheduling problem with different feasible modes of overlapping including rework and coordination/communication. The problem is formulated as a linear integer program. An example of a 30 activity project is provided to illustrate the utility and efficiency of the model. Our results highlight the closed interaction between resource constraints and overlapping modes and confirm the relevance of jointly consider them.

Keywords. Activity overlapping, concurrent engineering, project management, project scheduling.

Acknowledgements. This work was supported by the Jarislowsky Foundation, the SNC-Lavalin firm and by the Natural Sciences and Engineering Council of Canada (NSERC). This support is gratefully acknowledged.

---

* Corresponding author: Francois.Berthaut@cirrelt.ca

Dépôt légal – Bibliothèque et Archives nationales du Québec
Bibliothèque et Archives Canada, 2011

1 Introduction

The RCPSP (Resource-Constrained Project Scheduling Problem) has been addressed in numerous papers. Various models attempt to minimize project time completion while considering limited resources (Hartmann 1999). Hartmann and Briskorn (2010) have also presented an overview of different RCPSP extensions. Among extensions addressed in the literature, different practices have been developed to reduce time of project execution in order to establish a baseline schedule or to modify it following project delay during its execution through overlapping, crashing and substitution of activities (Gerk and Qassim 2008). In engineering project, overlapping is considered as “a core technique for saving development time” (Smith and Reinertsen 1998). It consists in starting a downstream activity before receiving all the final information required. It has been demonstrated to be a powerful tool for reducing product development times from conceptual design to production start-up in a wide range of industries, such as software (Blackburn and al. 1996), mobile phones (Lin et al 2010) automobiles (Clark and Fujimoto 1991) and airplanes (Sabbagh 1996). Overlapping has also been applied in design and build phases of construction projects (Pena-Mora and Li 2001; Dzeng 2006). Indeed, a common practice in construction projects is to reach 20% of build phase while design is completed at 80%. Overlapping activities or project phases, and the surrounding organizational activities required to support it, are often referred as simultaneous or concurrent engineering (Terwiesch and Loch 1999).

As preliminary information provided by upstream activities may evolve until it becomes final information, overlapping often causes additional rework and modification in downstream activities. Such rework may outweigh the overlap benefices of parallel activity execution in terms of cost and time, particularly if development uncertainty is not resolved early during the project (Terwiesch and Loch 1999). Indeed, if the development uncertainty is high, most of the tasks done on preliminary information will be reworked, which make overlapping unfavorable. Frequent information exchange between the development teams reduces the negative effect of overlapping, but requires additional time and cost for communication and coordination. However, project planners and managers determine overlapping strategies on an ad hoc basis without always considering rework and interaction between activities (Lin et al. 2010), yielding inefficient project management.

A growing body of literature in operations management has investigated the question of when and to which extent overlapping should be applied. Two groups of models have been developed in the literature to analyze overlapping interactions. First, many authors consider only a couple of activities or project phases and no resource constraints to establish the best trade-off between overlapping and rework. Krishnan et al. (1997) introduced the concept of information evolution and downstream sensitivity to describe interactions in overlapped activities with the assumption that communication and coordination are instantaneous and costless. Information evolution refers to the upstream generated information useful for downstream activities. Downstream sensitivity refers to the impact of a change in upstream activity on the downstream activity. They developed a model to determine when to start the downstream activity so as to minimize the development cycle time. Roemer et al. (2000) adapted the concepts of evolution and sensitivity to model the probability of rework as a function of the overlap duration and studied the time-cost tradeoffs in overlapping. Loch and Terwiesch (1998), who studied the time-to-market minimization problem, and Lin et al. (2010), who studied the time-cost tradeoffs problem, investigated the integrated problem of overlapping and information exchange policy assuming that information exchange usually requires time and cost. In addition to quantifying the amount of rework as a function of the overlap duration, these papers showed that the optimal overlapping strategy is to overlap as much as possible when information exchange is instantaneous and costless, while there exists an optimal overlap duration which differs from the maximum possible overlap duration when information exchange requires non-negligible time and cost.

Other approaches have considered whole projects instead of coupled of activities under the assumption that the relation between overlapping amount and rework is preliminary known for overlappable activities. They mostly use design structure matrix (DSM) to represent dependencies, to minimize feedbacks, and to identify overlapping opportunities between activities. Among other models, Gerk and Qassim (2008) developed an analytic project acceleration linear model via activity crashing, overlapping and substitution with resource constraints. Wang and Lin (2009) developed a stochastic overlapping process model to assess schedule risks. Their simulation model considers iterations and probabilities of rework. Iterations are mostly defined as interaction between design activities which lead to rework in upstream activities caused by feedbacks from downstream activities. However, their model does not take into account resource constraints. Cho and Eppinger (2005) introduced a simulation model with stochastic activity durations, overlapping, iterations, rework and considered resource constraints for some activities. They showed that theses constraints can delay some overlapped activities and delay the project. All these papers assume a simple linear relationship between rework and overlapping amount with an upper and lower bound and consider that information exchange is instantaneous and costless.

In summary, most contributions in the related literature fail to consider a realistic relationship between overlapping, rework and communication/coordination in the RCPSP. The objective of this paper is to extend the classical RCPSP with a realistic overlapping model that deals with additional workloads and costs incurred by rework and coordination/communication. We assume that the information flow is unidirectional from upstream to downstream activities. Consequently, the rework caused by overlapping is only assigned to the downstream activities and there is no activity iteration. The main difference with the aforementioned overlapping models is that overlapping is restricted to a set of feasible overlap durations for each couple of overlappable activities, instead of considering a continuous and bounded interval for the overlap duration. These overlapping modes are characterized by different overlap durations, rework durations and costs, and communication/coordination durations and costs. For convenience, the overlapping modes are subsequently converted into activity modes, each of which representing a combination of overlapping modes of an activity with the associated overlappable activities. This transformation enables to easily formulate the RCPSP with overlapping modes as a linear integer programming problem, which shares similarities with the classical multi-mode RCPSP model (Hartmann 1999). This model allows finding an optimal makespan in reasonable calculation time.

The remainder of the paper is organized as follows. Section 2 first describes the problem statement and assumptions. The gain maximization problem and the makespan minimization problem are formulated in Section 3. An illustrative example and computational results of the time-cost trade-offs are then presented in section 4. Section 5 concludes with recommendations for future work.

2 Problem Statement and assumptions

The project scheduling literature largely focuses on the generation of a precedence and resource feasible schedule that optimizes the schedule objective(s) (most often the project makespan or cost) and that should be used as a baseline schedule for executing the project. This schedule serves as a basis to allocation of resources, planning of material procurement, communication and coordination within the projects and with external entities (client, subcontractors,…), etc. Here, we assume that all information required for the scheduling of the project is known in advance, and consequently the problem is formulated and solved in a deterministic environment. This section is then devoted to present the project model and the information required to solve the project scheduling problem with resource constraints and overlappable activities.

A project is defined by a set of activities, S, including two fictitious activities 0 and $n+1$, which correspond to the project start and project end, respectively, with zero processing time. We denote by d_j the estimated nominal processing time of activity j considering that all the final information required from preceding activities are available at its start; in other word, if activity j is processed without overlapping. All the symbols and their definitions used along this paper are presented in Table 1.

2.1 Precedence constraints

Frequently used project-planning methods provide graphic descriptions of task workflows in the form of the so-called activity-on-node or activity-on-arc networks. These networks depict the logical execution sequence of dependent (sequential) activities and independent (parallel or concurrent) activities. However, these tools fail to incorporate interdependent-type relation, activities' iterations and to model information flows between activities.

The Design Structure Matrix (DSM) representation can handle these additional relations between activities with the broader concept of information sharing (Browning 2001). Information exchange between activities can occur at the beginning, the middle or the end of an activity and includes both tangible and intangible types such as parts, part dimensions, and bill of materials, which constitute the outputs from an upstream activity and are required to begin the work of a downstream activity. A DSM is a square matrix where rows and columns represent activities. It aims to represent the information flows for a given subset of activities and constitutes the first step in analyzing potential feedbacks. Feedback information exchanges from downstream to upstream activities correspond to design modification requests due to inability to meet target design requirements or design flaws detected in downstream stages (Wang and Lin 2009). Any feedback information exchange from downstream activities lead to modifications and reworks performed by

Table 1 Symbols and definitions

Symbol	Definition	Symbol	Definition
S	Set of activities	Cr_{ijm}	Rework cost in the downstream activity j when (i, j) are overlapped in precedence mode m
n	Number of non-dummy activities	Cc_{ijm}	Total cost for coordination and communication when (i, j) are overlapped in precedence mode m
E	Set of temporal or precedence constraints	p_j	Number of execution modes of the activity j
$i \rightarrow j$ (i, j)	Precedence constraint	m_{ijp}	Precedence mode of the couple (i, j) in execution mode p (of activity j)
d_j	Processing time of activity j	β_{ijp}	Amount of overlap duration between activities i and j in execution mode p (of activity i)
A	Set of couples of overlappable activities	μ_{ijp}	Expected amount of rework in activity j in execution mode p (of activity i)
P	Set of couples of non-overlappable activities	ρ_{ijp}	Expected total amount of time for coordination and communication when (i, j) are overlapped in execution mode p (of activity i)
Po_j	Set of immediate predecessors of activity j that are overlappable with activity j	m'_{ijp}	Precedence mode of the couple (i, j) in execution mode p (of activity i)
Pn_j	Set of immediate predecessors of activity j that are not overlappable with activity j	β'_{ijp}	Amount of overlap duration between activities i and j in execution mode p (of activity j)
P_j	Set of immediate predecessors of activity j	μ'_{ijp}	Expected amount of rework in activity j in execution mode p (of activity j)
So_j	Set of immediate successors of activity j that are overlappable with activity j	ρ'_{ijp}	Expected total amount of time for coordination and communication when (i, j) are overlapped in execution mode p (of activity j)
Sn_j	Set of immediate successors of activity j that are not overlappable with activity j	μ_{jp}	Expected total amount of rework in activity j in execution mode p
S_j	Set of immediate successors of activity j	δ_{jp}	Expected total amount of time for coordination and communication in activity j in execution mode p
R	Set of renewable resources	CR_{jp}	Rework cost of activity j in execution mode p
R_k	Constant amount of available units of renewable resource k	CC_{jp}	Total cost for coordination and communication of activity j in execution mode p
R_{jk}	Per period usage of activity j of renewable resource k	Co	Opportunity cost (cost of increasing/decreasing the makespan by one unit of time)
m_{ij}	Number of precedence modes of the couple (i, j)	D	Project Due date
α_{ij}	Amount of overlap duration between activities i and j	T	Upper bound of the project makespan
r_{ij}	Expected amount of rework in the downstream activity j when (i, j) are overlapped	C_{lim}	Upper bound of the total overlapping cost
σ_{ij}	Expected total amount of time for coordination and communication when (i, j) are overlapped	$t = 0,..,T$	Periods
α_{ijm}	Amount of overlap duration between activities i and j in precedence mode m, expressed as a fraction of d_j	EF_j	Earliest possible finish time of activity j
r_{ijm}	Expected amount of rework in the downstream activity j when (i, j) are overlapped in precedence mode m	LF_j	Latest possible finish time of activity j
σ_{ijm}	Expected total amount of time for coordination and communication when (i, j) are overlapped in precedence mode m		

the upstream activities to accommodate these changes, and iterations between upstream and downstream activities can virtually occur to fix the problems identified. In order to minimize feedbacks, the DSM can be partitioned using block triangularization algorithm to obtain a unidirectional sequence of information exchange (Browning 2001). As a last resort, activities can be aggregated or decomposed into lower-level activities to eliminate feedbacks.

In this paper, we assume that such preliminary studies have been conducted to identify the nature of relations between activities and to determine a feasible sequence of activities without any feedback from downstream activities. The project is then only composed of independent and dependent activities and the resulting information flow within the project between activities is assumed to be unidirectional from upstream to downstream activities.

The analysis of information exchanges between dependent couple activities enables to categorize them into non-overlappable and overlappable ones. The former represents the case where a downstream activity requires the final output information from an upstream activity to be executed or the completion of the upstream activity. The latter represents the case where a downstream activity can begin with preliminary information and receives final update at the end of the upstream activity. This relation provides the opportunity to overlap two activities so that a downstream activity can start before an upstream activity is finished. While the non-overlappable activities are connected with the classical finish-to-start precedence constraint, the overlappable activities are connected with a finish-to-start-plus-lead time precedence constraint where the lead-time accounts for the amount of overlap. Note that the finish-to-start precedence constraint is the most conventional type of relationship used in practice and in project management tools such as MS Project or Primavera (Cho and Eppinger 2005).

In the remainder of the paper, we denote by A and P the sets of couples of overlappable and non-overlappable activities, respectively. Similarly, for each activity j, Po_j and Pn_j represent the set of overlappable and non-overlappable predecessors, respectively, while So_j and Sn_j denote the set of overlappable and non-overlappable succesors, respectively. The set of precedence constraints in the project, E, and the set of immediate predecessors, P_j, and immediate successors, S_j, of each activity j are defined by:

$$E = A \cup P \quad (1)$$

$$P_j = Po_j \cup Pn_j\,,\; S_j = So_j \cup Sn_j\,,\quad \forall j \in S \quad (2)$$

2.2 Model of the overlapping process

Figure 1 shows the overlapping process of two activities (i, j) in A. The downstream activity j starts with preliminary inputs from the upstream activity i. The amount of overlap, α_{ij}, is expressed as a fraction of the downstream activity's duration. As the upstream activity proceeds, its information evolves to its final form and is released to the downstream activity j at its completion. This approach implies that the traditional pattern of exchange of finalized information at the end of the upstream activity is altered to a more frequent exchange of evolving information during the overlapping process. However, additional rework is often necessary to accommodate the changes in the upstream information in the downstream development. The expected duration of the sum of rework is denoted by r_{ij}. Moreover, frequent information exchange allows the downstream team to be aware of the latest upstream change to be incorporated in their work. For each time information

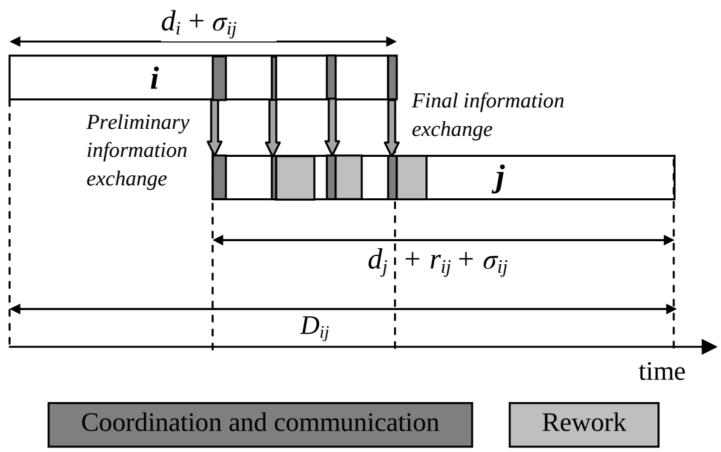

Fig 1. Overlapping process of two activities

is exchanged, upstream and downstream teams have to drop everything they are doing and commit themselves to set a cross-functional team meeting and discuss the latest changes for downstream incorporation (Loch and Terwiesch 1998; Lin et al. 2010). The expected duration of the sum of information exchange durations for communication and coordination due to overlapping is denoted by σ_{ij}. The total amount of time required to execute both activities, D_{ij}, is expressed as follows:

$$D_{ij} = d_i + d_j \cdot (1 - \alpha_{ij}) + r_{ij} + \sigma_{ij} \quad (3)$$

If $d_j \geq d_i$, the amount of overlap is usually bounded by the fraction d_i / d_j in order to prevent the downstream activity to start before the upstream activity. If overlapping was not applied, the total amount of time required to execute both activities would simply be $D_{ij} = d_i + d_j$. Depending on the nature of the activities, there may exist a trade-off between time gains from overlapping, rework, and communication and coordination. In addition, if we consider the costs associated with rework, communication and coordination, and the opportunity costs for finishing the project earlier (premium) or later (penalty for delay), there may also exists a trade-off between additional cost for overlapping and opportunity cost for finishing earlier. These give rise to the three following main optimization problems proposed in literature for the overlapping of two activities without resource constraints, with or without the assumption of instantaneous and costless information exchange:

- time-to-market minimization problem, with or without a maximum costs constraint (Loch and Terwiesch 1998; Roemer et al. 2000; Lin et al. 2010),
- cost minimization problem, subject to a maximum time-to-market constraint (Roemer et al. 2000; Roemer an Ahmadi 2004),
- gain maximization problem, subject to a maximum time-to-market constraint (Roemer et al. 2000; Lin et al. 2009; Lin et al. 2010).

The main issue with the overlapping problem is to quantify the amount of rework as a function of the amount of overlap. Indeed, the overlapping problem requires exploring the behavior and interaction of activities during their processes. Krishnan et al. (1997) presented a pioneer paper in this field. They proposed a model of dependency based on the upstream information evolution, which characterizes the refinement of information from its preliminary form to a final value, and the downstream sensitivity, which represents the duration of a downstream iteration to incorporate upstream changes. Roemer et al. (2000) and Roemer and Ahmadi (2004) introduced the concept of probability of rework as a function of the overlap duration, which encompasses both the evolution and sensitivity models proposed in Krishnan and al. (1997). When information exchange requires non-negligible time and cost, Loch and Terwiesch and Lin et al. (2010) assumed that each information exchange (i.e., meeting) has a setup time and a setup cost considered as constant. The communication/coordination policy is then characterized by the frequency and the number of information exchange. Loch and Terwiesch (1998) adapted the concepts proposed by Krishnan et al. (1997) by considering the upstream evolution as the rate of modifications in the upstream and the downstream sensitivity as the impact of a modification on downstream rework, and jointly analyzed overlapping and communication policies between two activities. Lin and al. (2010) investigated the evolution of the downstream progress and refined the model proposed by Loch and Terwiesch (1998).

An important finding of the aforementioned papers is that the duration of rework is a convex increasing function of the amount of overlap. The former statement is intuitive: if the amount of overlap increases, then the preliminary information at the downstream activity's start will be more unreliable and more downstream changes must be incorporated. In addition, when information exchange is considered, the optimal coordination and communication policy for a given amount of overlap is such that the resulting duration of coordination and communication is concave or convex depending on the shape of the upstream information evolution (Loch and Terwiesch 1998). When the upstream evolution is linear, the duration coordination and communication is a non-decreasing function with respect to the amount of overlap (Loch and Terwiesch 1998; Lin et al. 2010).

Another important finding of the aforementioned papers is that the time to complete the upstream and downstream activities is a convex increasing function of the amount of overlap when information exchange has negligible cost and duration. Therefore, the optimal amount of overlap is the maximum feasible amount of overlap. With non-negligible information exchange cost and duration, the time to complete the upstream and downstream activities is either convex, concave or concave-convex with respect to the amount of overlap depending on the shape of the upstream information evolution (Loch and Terwiesch 1998). In particular, the time to complete the upstream and downstream activities is convex when the upstream evolution is linear, such that the optimal amount of overlap may be greater that the maximum feasible amount of overlap (Loch and Terwiesch 1998; Lin et al. 2010).

To sum up, the amount of overlap which minimizes the time to complete the upstream and downstream activities has been derived under different conditions and lead to the conclusion that the optimal overlap amount is not necessarily the maximum feasible amount of overlap and that the time to complete the upstream and downstream activities is either convex, concave or concave-convex with respect to the amount of overlap. However, when overlapping is considered for more than two activities for a whole project with several overlappable couples of activities, the models of overlapping process proposed in the current literature consider a simplistic linear relation between the rework and the amount of overlap. We propose in the next section an overlapping model for project with several overlappable couples of activities, which both relaxes this assumption and encompasses any overlapping process proposed so far for two activities.

The overlapping costs have also received much attention in the literature in order to formulate the cost minimization and the gain maximization problems. The overlapping costs are composed of the cost of rework and the communication/coordination cost. The cost of rework are usually considered as a linear function of the rework duration with or without a fixed cost (Roemer et al. 2000; Roemer and Ahmadi 2004; Gerk and Qassim 2008; Lin et al. 2010). For example, Roemer et al. (2000) argued that the rework cost corresponds to the hours of engineering spend on rework multiplied with the average wage of engineers per unit of time. Similarly, Lin et al. (2010) assumed that communication and coordination has a constant setup cost per meeting or information exchange. Therefore, the communication/coordination cost is proportional to the amount of time spent for communication/coordination.

2.3 Precedence and overlapping modes

In order to study the interaction between overlapping and resource constraints in the scheduling optimization problem with multiple activities including several overlapping opportunities, the relations between the amount of overlap, rework duration and cost and communication/coordination duration and cost are required for a range of amount of overlap for each couple of overlappable activities. Indeed, the optimal overlap amounts for a resource-constraints project composed of several couples of overlappable activities are not necessarily set to the optimal values found for each couple of activities (Browning and Eppinger 2002; Cho and Eppinger 2005; Gerk and Qassim 2008).

In this paper, overlapping is assumed to be defined for discrete values of overlap durations. First, this assumption is more realistic considering that scheduling is performed in practice on a period-by-period basis (i.e., hour, day, week): resource availabilities and allocations are estimated per period, while activity durations are discrete multiples of one period (Hartmann 1999). Second, activity progress is measured in practice according to the completion of internal milestones which corresponds to important events, such as design criteria frozen, detailed design completed, drawings finalized, or any activity deliverables. This preliminary information is issued at intermediate points and used as input for a downstream activity. Therefore, the start time of an overlapped downstream activity is restricted to a finite number of instants corresponding to upstream activities' milestones which constitutes different feasible modes for the execution of overlapping activities. Each overlapping mode is characterized by an amount of overlap expressed as a fraction of the downstream activity's duration, rework duration and cost and communication/coordination duration and cost. These parameters can be either derived from models of overlapping process presented in previous section when historical data are available or estimated by engineers for each overlapping mode.

Table 2a Precedence modes for a non-overlappable couple of activities (i, j) in P

Overlapping mode of couple (i, j), m	Amount of overlap, α_{ijm}	Rework duration, r_{ijm}	Coordination/ communication duration, σ_{ijm}	Rework cost, Cr_{ijm}	Coordination/ communication cost, Cc_{ijm}
1	0	0	0	0	0

Table 2b Overlapping modes for an overlappable couple of activities (i, j) in A

Overlapping mode of couple (i, j), m	Amount of overlap, α_{ijm}	Rework duration, r_{ijm}	Coordination/ communication duration, σ_{ijm}	Rework cost, Cr_{ijm}	Coordination/ communication cost, Cc_{ijm}
1	0	0	0	0	0
2	α_{ij2}	r_{ij2}	σ_{ij2}	Cr_{ij2}	Cc_{ij2}
⋮	⋮	⋮	⋮	⋮	⋮
m_{ij}	$\alpha_{i,j,mij}$	$r_{i,j,mij}$	$\sigma_{i,j,mij}$	$Cr_{i,j,mij}$	$Cc_{i,j,mij}$

Overlapping modes can be generalized to precedence modes in order to describe all precedence relationships between activities. For each couple of precedence constraints $i \rightarrow j$, there exists at least one precedence mode which corresponds to a basic finish-to start relation without overlapping. When $(i, j) \in A$, there exist additional precedence modes associated with the different overlapping strategies. The precedence modes can be expressed as presented in Tables 2a and 2b. When activities i and j are overlappable, they can be either overlapped and executed in mode $m = 2,\ldots,m_{ij}$, or sequentially performed in mode $m = 1$ without overlapping. As depicted in Figure 2, it is important to note that the precedence constraints on the finish time of activities i and j will defer depending on the overlapping mode: when not overlapped, the downstream activity start time is superior or equal to the upstream activity finish time, whereas the downstream activity start time is equal to the upstream activity finish time minus one of the feasible overlap duration in the case of overlapping.

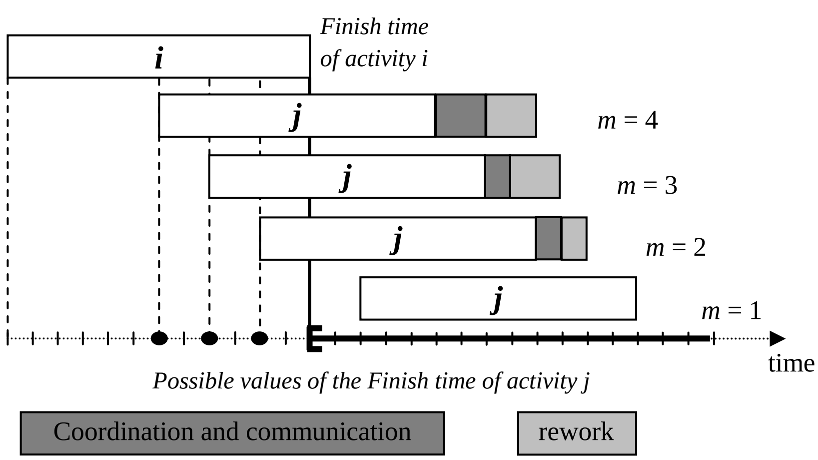

Fig. 2 Precedence constraints on the finish times of two overlappable activities i and j depending on the overlapping modes m

2.4 Multiple overlapping and activity modes

We assume that there is no restriction concerning the number of overlappable or non-overlappable predecessors. If an activity is overlapped by multiple upstream activities, feasible overlapping modes are assumed to be compatible. Consider for example the case of a downstream activity j with two upstream activities, denoted by i_1 and i_2. If both couples (i_1, j) and (i_2, j) are overlapped, the amount of rework in downstream activity is between the maximum of single rework and the sum of them, depending on the duplicate rework, as stated in Cho and Eppinger (2005). Without loss of generality, the latter is considered in the model. Similarly the amount of time spent for communication/coordination in activity j is assumed to be the sum of the communication/coordination durations with its overlapped predecessors and successors.

In typical projects involving engineering phases, the number of precedence and overlapping relationships may largely exceeds the number of activities. As each activity can have several overlappable or non-overlappable predecessors and successors, we introduce the notion of execution modes associated to activities. Each activity mode represents a combination of possible precedence or overlapping modes of an activity with its overlappable or non-overlappable predecessors and successors. Consequently, the set of activity modes {1,…, p_j} for each activity is generated by a full factorial design of the precedence and overlapping modes with its predecessors and successors. Table 3a shows the activity modes in the case of non-overlappable predecessors and successors. Similarly, Tables 3b and 3c presents the activity modes in the case of only one overlappable predecessor (with four overlapping modes) and no overlappable successor, and one overlappable predecessor and one overlappable successor (each with three overlapping modes), respectively.

Table 3a Activity modes of activity j in the case of non-overlappable predecessors and successors

p	$\forall i \in Pn_j$				$\forall k \in Sn_j$				μ_{jp}	δ_{jp}	CR_{jp}	CC_{jp}
	m_{ijp}	β_{ijp}	μ_{ijp}	ρ_{ijp}	m'_{jkp}	β'_{jkp}	μ'_{ipk}	ρ'_{jkp}				
1	1	0	0	0	1	0	0	0	0	0	0	0

Table 3b Activity modes of activity j in the case of one overlappable predecessor (with 4 overlapping modes) and no overlappable successors

p	$\forall i \in Po_j$				$\forall i' \in Pn_j$				$\forall k \in Sn_j$				μ_{jp}	δ_{jp}	CR_{jp}	CC_{jp}
	m_{ijp}	β_{ijp}	μ_{ijp}	ρ_{ijp}	$m_{i'jp}$	$\beta_{i'jp}$	$\mu_{i'jp}$	$\rho_{i'jp}$	m'_{jkp}	β'_{jkp}	μ'_{ipk}	ρ'_{jkp}				
1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
2	2	α_{ij2}	r_{ij2}	σ_{ij2}	1	0	0	0	1	0	0	0	r_{ij2}	σ_{ij2}	Cr_{ij2}	Cc_{ij2}
3	3	α_{ij3}	r_{ij3}	σ_{ij3}	1	0	0	0	1	0	0	0	r_{ij3}	σ_{ij3}	Cr_{ij3}	Cc_{ij3}
4	4	α_{ij4}	r_{ij4}	σ_{ij4}	1	0	0	0	1	0	0	0	r_{ij4}	σ_{ij4}	Cr_{ij4}	Cc_{ij4}

Table 3c Activity modes of activity *j* in the case of one overlappable predecessor (with 3 overlapping modes) and one overlappable successor (with 3 overlapping modes)

p	$\forall i \in Po_j$				$\forall i' \in Pn_j$				$\forall k \in So_j$				$\forall k' \in Sn_j$				μ_{jp}	δ_{jp}	CR_{jp}	CC_{jp}
	m_{ijp}	β_{ijp}	μ_{ijp}	ρ_{ijp}	$m_{i'jp}$	$\beta_{i'jp}$	$\mu_{i'jp}$	$\rho_{i'jp}$	m'_{jkp}	β'_{jkp}	μ'_{ipk}	ρ'_{jkp}	$m'_{jk'p}$	$\beta'_{jk'p}$	$\mu'_{jk'p}$	$\rho'_{jk'p}$				
1	1	0	0	0	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
2	1	0	0	0	1	0	0	0	2	α_{jk2}	r_{jk2}	σ_{jk2}	1	0	0	0	r_{jk2}	σ_{jk2}	0	$\frac{Cc_{jk2}}{2}$
3	1	0	0	0	1	0	0	0	3	α_{jk3}	r_{jk3}	σ_{jk3}	1	0	0	0	r_{jk3}	σ_{jk3}	0	$\frac{Cc_{jk3}}{2}$
4	2	α_{ij2}	r_{ij2}	σ_{ij2}	1	0	0	0	1	0	0	0	1	0	0	0	r_{ij2}	σ_{ij2}	Cr_{ij2}	$\frac{Cc_{ij2}}{2}$
5	2	α_{ij2}	r_{ij2}	σ_{ij2}	1	0	0	0	2	α_{jk2}	r_{jk2}	σ_{jk2}	1	0	0	0	$r_{ij2} + r_{jk2}$	$\sigma_{ij2} + \sigma_{jk2}$	Cr_{ij2}	$\frac{Cc_{ij2} + Cc_{jk2}}{2}$
6	2	α_{ij2}	r_{ij2}	σ_{ij2}	1	0	0	0	3	α_{jk3}	r_{jk3}	σ_{jk3}	1	0	0	0	$r_{ij2} + r_{jk3}$	$\sigma_{ij2} + \sigma_{jk3}$	Cr_{ij2}	$\frac{Cc_{ij2} + Cc_{jk3}}{2}$
7	3	α_{ij3}	r_{ij3}	σ_{ij3}	1	0	0	0	1	0	0	0	1	0	0	0	r_{ij3}	σ_{ij3}	Cr_{ij3}	Cc_{ij3}
8	3	α_{ij3}	r_{ij3}	σ_{ij3}	1	0	0	0	2	α_{jk2}	r_{jk2}	σ_{jk2}	1	0	0	0	$r_{ij3} + r_{jk2}$	$\sigma_{ij3} + \sigma_{jk2}$	Cr_{ij3}	$\frac{Cc_{ij3} + Cc_{jk2}}{2}$
9	3	α_{ij3}	r_{ij3}	σ_{ij3}	1	0	0	0	3	α_{jk3}	r_{jk3}	σ_{jk3}	1	0	0	0	$r_{ij3} + r_{jk3}$	$\sigma_{ij3} + \sigma_{jk3}$	Cr_{ij3}	$\frac{Cc_{ij3} + Cc_{jk3}}{2}$

m_{ijp}, β_{ijp}, μ_{ijp} and ρ_{ijp} denote the precedence/overlapping mode, the amount of overlap, the rework duration and the communication/coordination duration of the couple (i, j) in activity mode $p = 1,\ldots, p_j$ (i.e., the activity modes of the downstream activity j), respectively. The same symbols with the prime symbol represent the same definitions express in activity modes of the upstream activity i. δ_{jp}, μ_{jp}, CR_{jp} and CC_{jp} are the total communication/coordination duration (i.e., the sum of communication/coordination duration with overlappable predecessors and successors), the total rework duration, the total rework cost and the total communication/coordination cost of activity j in mode $p = 1,\ldots,p_j$. Note that the communication/coordination cost associated with a couple (i, j) in A must be split between activity i and j. Without loss of generality, the communication/coordination cost is assumed to be equally split between upstream and downstream activities.

3 Performance optimization models

Each activity j in S must finish within the time window $\{EF_j,\ldots,LF_j\}$ with respect to the precedence relations, the overlapping opportunities and the activity durations. As stated in Hartmann (1999), they can be derived from the traditional forward recursion and backward recursion algorithms considering that the project must start at time 0 and that T constitutes an upper bound of the project's makespan (i.e., the sum of processing times of all activities). We define the decision variables (i.e., the finish times and the overlapping modes) as follows:

$$X_{jtp} = \begin{cases} 1 & \text{if activity } j \text{ is executed in mode } p \text{ and finished at time } t \\ 0 & \text{otherwise} \end{cases}$$

$$\forall j \in S,\ \forall t \in [0,T] \text{ and } \forall p \in [1, p_j] \tag{4}$$

The decision on the activity modes can be classed into three cases. On the one hand, if activity *j* is not overlappable with any immediate predecessor or successor, the decision is simply not to overlap. On the other hand, if activity *j* is overlappable, this activity can be either overlapped with at least one of its overlappable predecessor or successor ($p > 1$) or not overlapped ($p = 1$). The resource-constrained scheduling problem with overlapping is formulated in this section with the objective of maximizing the project gain. Next, we present a variation of this problem with the objective of minimizing the project makespan.

3.1 Project gain maximization problem

The resource-constrained scheduling problem of maximizing the project gain with overlapping can be formulated with the nonlinear 0-1 integer programming model as follows:

$$\text{Maximize } Co \cdot \left(D - \sum_{p=1}^{p_{n+1}} \sum_{t=EF_{n+1}}^{LF_{n+1}} t \cdot X_{n+1,t,p} \right) - \sum_{j=2}^{n} \sum_{p=1}^{p_j} \sum_{t=EF_j}^{LF_j} \left(CR_{jp} + CC_{jp} \right) \cdot X_{jtp} \tag{5}$$

Subject to

If $\sum_{t=EF_j}^{LF_j} X_{jt1} = 1$ then

$$\sum_{p=1}^{p_i} \sum_{t=EF_i}^{LF_i} t \cdot X_{itp} \le \sum_{t=EF_j}^{LF_j} \left(t - d_j \right) \cdot X_{jt1},\ \forall j \in S,\ \forall i \in P_j \tag{6}$$

If $\sum_{p=2}^{p_j} \sum_{t=EF_j}^{LF_j} X_{jtp} = 1$ then

$$\sum_{p=1}^{p_i} \sum_{t=EF_i}^{LF_i} \left(t - \rho_{ijp} \right) \cdot X_{itp} = \sum_{p=1}^{p_j} \sum_{t=EF_j}^{LF_j} \left[t - d_j \cdot \left(1 - \beta_{ijp} \right) - \left(\mu_{jp} + \delta_{jp} \right) \right] \cdot X_{jtp},\ \forall j \in S,\ \forall i \in Po_j \tag{7}$$

$$\sum_{j=2}^{n} \left[R_{jk} \cdot \left(\sum_{p=1}^{p_j} \sum_{b=t}^{t+d_j-1+\mu_{jp}+\delta_{jp}} X_{jbp} \right) \right] \le R_k,\ \forall k \in R,\ \forall t \in [0,T] \tag{8}$$

$$\sum_{p=1}^{p_i} \sum_{t=EF_i}^{LF_i} t \cdot X_{itp} \le \sum_{p=1}^{p_j} \sum_{t=EF_j}^{LF_j} t \cdot X_{jtp},\ \forall j \in S,\ \forall i \in Po_j \tag{9}$$

$$\sum_{p=1}^{p_i}\sum_{t=EF_i}^{LF_i}\beta'_{ijp}\cdot X_{itp}=\sum_{p=1}^{p_j}\sum_{t=EF_j}^{LF_j}\beta_{ijp}\cdot X_{jtp}\ ,\ \forall j\in S\ ,\forall i\in Po_j \tag{10}$$

$$\sum_{p=1}^{p_j}\sum_{t=EF_j}^{LF_j}X_{jtp}=1\ ,\forall j\in S \tag{11}$$

$$X_{jtp}=\{0,1\}\ ,\forall j\in S\ ,\ \ \forall t\in[0,T]\ \text{and}\ \forall p\in[1,p_j] \tag{12}$$

The objective function (5) maximizes the project gain, which is composed of the opportunity cost for finishing earlier or later than a target time, *D* (i.e., the project makespan without overlapping or any project due date), and the overlapping costs (rework and communication/coordination costs). Constraints (6) represent the finish-to-start precedence constraints when activities are not overlapped. If activities are overlapped, constraints (7) state that the downstream activity must start at the upstream activity finish time minus one of the feasible overlap duration. Constraints (6) and (7) reflect the precedence and overlapping constraints presented in Figure 2. Constraints (8) define the resource constraints. Constraints (9) guarantee that the downstream activity of a couple of overlappable activities can not finish before the upstream activity's finish time. Constraints (10) state that the activity modes of two overlappable activities are such that the overlapping modes are the same for both activities. Constraints (11) ensure that each activity is associated with one activity mode and one finish time. Finally, constraints (12) define the aforementioned binary decision variables.

The nonlinear 0-1 integer non-linear programming model given by the objective function (5) and the constraints (6)-(12) can be transformed into a linear 0-1 integer programming model. Constraints (6) and (7) are reformulated as follows:

$$\sum_{p=1}^{p_i}\sum_{t=EF_i}^{LF_i}\left(t-\rho_{ijp}\right)\cdot X_{itp}\le\sum_{p=1}^{p_j}\sum_{t=EF_j}^{LF_j}\left[t-d_j\cdot\left(1-\beta_{ijp}\right)-\left(\mu_{jp}+\delta_{jp}\right)\right]\cdot X_{jtp}\ ,\forall j\in S\ ,\ \ \forall i\in P_j \tag{13}$$

$$\sum_{p=1}^{p_j}\sum_{t=EF_j}^{LF_j}\beta_{ijp}\cdot X_{itp}\le Y_{ij}\ ,\forall j\in S\ ,\ \forall i\in Po_j \tag{14}$$

$$\sum_{p=1}^{p_i}\sum_{t=EF_i}^{LF_i}\left(t-\rho_{ijp}\right)\cdot X_{itp}\ge\sum_{p=1}^{p_j}\sum_{t=EF_j}^{LF_j}\left[t-d_j\cdot\left(1-\beta_{ijp}\right)-\left(\mu_{jp}+\delta_{jp}\right)\right]\cdot X_{jtp}-T\cdot\left(1-Y_{ij}\right),$$
$$\forall j\in S\ ,\ \forall i\in Po_j \tag{15}$$

$$Y_{ij}=\{0,1\}\ ,\forall j\in S\ ,\ \forall i\in Po_j \tag{16}$$

Note that Y_{ij} is an additional binary variable. Constraints (13) represent the finish-to-start precedence constraints, with a negative lead time in the case of overlapping. According to constraints (14), if two overlappable activities (*i*, *j*) are overlapped, then Y_{ij} = 1 and thus the union of constraints (13) and (15) is equivalent to the equality constraints (7). If activities (*i*, *j*) are not overlapped, then Y_{ij} is unrestricted and constraints (15) are not restrictive.

The project gain maximization problem can then given be formulated as a linear 0-1 integer programming problem given by the objective function (5) and the constraints (8)-(16).

3.2 Project makespan minimization problem

The project makespan minimization problem can be formulated by replacing the objective function (5) by the following objective function:

$$\text{Minimize } \sum_{p=1}^{p_{n+1}} \sum_{t=EF_{n+1}}^{LF_{n+1}} t \cdot X_{n+1,t,p} \tag{17}$$

The objective function (17) minimizes the finish time of the dummy sink activity and therefore, the project's makespan. In addition, we introduce the following constraint to ensure that the total overlapping cost will not exceed a given value:

$$\sum_{j=2}^{n} \sum_{p=1}^{p_j} \sum_{t=EF_j}^{LF_j} \left(CR_{jp} + CC_{jp}\right) \cdot X_{jtp} \leq C_{lim} \tag{18}$$

The project makespan minimization problem can then be formulated as a linear 0-1 integer programming problem given by the objective function (17) and the constraints (8)-(16) and (18).

The project gain maximization and makespan minimization problems allow to study the time-cost trade-offs between project duration and overlapping costs with resource constraints. To our knowledge, such scheduling problems with resource constraints and overlapping opportunities has only been treated by Gerk and Qassim (2008). However, these authors used a simplified overlapping model with a linear relation between rework and overlap duration and they assumed that information exchange is instantaneous and costless. Our formulation allow any overlapping process to be applied, with the condition that overlapping can only be executed at feasible modes.

4 Illustrative example

4.1 Project data

We consider a project instance generated by Kolisch and Sprecher (1996) composed of 30 non-dummy activities and 4 renewable resources. The activity durations, resource consumptions and precedence relations are summarized in Table 4. The availability of the resources are set to $R_k = 20$, $k = 1,\ldots,4$. As no overlapping was defined in the original instance, the additional overlapping data have been generated. Fifteen couples of overlappable activities and their respective overlapping modes have been considered, as depicted in Table 5. The number of overlapping mode has been restricted to 3 for each couple of overlappable activities. As assumed in Roemer et al. (2000), Roemer and Ahmadi (2004) and Lin et al. (2010), the rework and

Table 4 Project data composed of 30 non-dummy activities

Activit j	Duration d_j	Resource consumption R_{j1}	R_{j2}	R_{j3}	R_{j4}	Set of non-overlappable predecessors, Pn_j	Set of overlappable predecessors, Po_j
1	0	0	0	0	0	{}	{}
2	14	1	6	0	0	{1}	{}
3	5	5	2	0	4	{}	{2}
4	5	0	0	0	3	{3}	{}
5	12	4	8	6	8	{2}	{}
6	11	0	7	0	0	{}	{3}
7	5	7	0	0	8	{3}	{}
8	6	0	0	6	10	{}	{7}
9	8	6	3	8	0	{2}	{}
10	12	0	1	3	6	{}	{8}
11	14	7	0	7	0	{4,8}	{}
12	15	0	0	0	8	{5}	{10}
13	13	0	0	2	5	{}	{10}
14	15	5	4	0	7	{}	{11}
15	5	4	0	0	0	{4, 7, 9}	{}
16	5	2	0	0	0	{13}	{}
17	10	1	0	0	3	{10}	{11}
18	12	4	1	0	4	{4}	{6}
19	12	0	0	10	3	{13, 15, 18}	{}
20	9	4	6	0	0	{7, 18}	{}
21	7	8	2	9	0	{}	{13, 20}
22	6	3	0	7	0	{8, 15}	{}
23	14	0	0	0	6	{5, 9, 21}	{}
24	7	8	0	0	0	{12, 19, 23}	{}
25	12	8	0	0	4	{16, 17}	{24}
26	8	0	8	8	0	{9}	{}
27	7	0	1	3	7	{14, 17, 21}	{}
28	15	4	0	0	2	{15, 26, 27}	{}
29	15	0	8	0	6	{11, 21}	{19}
30	8	3	0	7	8	{5, 29}	{28}
31	8	0	0	0	5	{22, 30}	{25}
32	0	0	0	0	0	{31}	{}

communication/coordination costs are considered as linear functions of the time spent on rework and communication, where the linear factors are the average wages of the teams per unit time (i.e., $200 per unit time for each resource). As a reminder, the overlap amount, the rework and the communication/coordination data for non-overlappable activities are set to zero. For the sake of conciseness, the activity modes of each activity are not all presented in this paper, as they are easily derived from Table 5. We only present in Table 6 the activity modes of activity 8 as an example. Finally, the opportunity cost is set to $5000 per unit time, while the project due date is set to the makespan found without overlapping modes.

Table 5 Overlapping data for the couples of overlappable activities

activities (i, j)	mode m	α_{ijm}	r_{ijm}	σ_{ijm}	Cr_{ijm} ($)	Cc_{ijm} ($)	activities (i, j)	mode m	α_{ijm}	r_{ijm}	σ_{ijm}	Cr_{ijm} ($)	Cc_{ijm} ($)
(2, 3)	1	0	0	0	0	0	(11, 17)	1	0	0	0	0	0
	2	0.4	0	0	0	0		2	0.3	1	0	800	0
	3	0.8	1	0	2200	0		3	0.6	2	1	1600	3600
(3, 6)	1	0	0	0	0	0	(13, 21)	1	0	0	0	0	0
	2	2/11	1	0	1400	0		2	2/7	0	0	0	0
	3	4/11	1	1	1400	3600		3	4/7	2	1	7600	5200
(6, 18)	1	0	0	0	0	0	(19, 29)	1	0	0	0	0	0
	2	5/12	1	1	1800	3200		2	4/15	1	0	2800	0
	3	8/12	3	2	5400	6400		3	8/15	2	1	5600	5400
(7, 8)	1	0	0	0	0	0	(20, 21)	1	0	0	0	0	0
	2	1/3	1	0	3200	0		2	3/7	1	0	3800	0
	3	2/3	2	1	6400	6200		3	5/7	2	1	7600	5800
(8, 10)	1	0	0	0	0	0	(24, 25)	1	0	0	0	0	0
	2	0.25	1	0	2000	0		2	0.25	1	1	2400	4000
	3	5/12	2	0	4000	0		3	5/12	2	1	4800	4000
(10, 12)	1	0	0	0	0	0	(25, 31)	1	0	0	0	0	0
	2	4/15	1	0	1600	0		2	0.25	1	0	1000	0
	3	8/15	2	2	3200	7200		3	0.625	2	1	2000	3400
(10, 13)	1	0	0	0	0	0	(28, 30)	1	0	0	0	0	0
	2	4/13	1	0	1400	0		2	0.375	1	0	3600	0
	3	9/13	2	2	2800	6800		3	0.75	1	1	3600	4800
(11, 14)	1	0	0	0	0	0							
	2	1/3	2	0	6400	0							
	3	0.6	3	2	9600	12000							

Table 6 Activity modes of activity 8

p	Overlappable predecessor, i = 7				Overlappable successor, k = 10				Non-overlappable successors, k' = 11, 22				μ_{8p}	δ_{8p}	CR_{8p} ($)	CC_{8p} ($)
	m_{78p}	β_{78p}	μ_{78p}	ρ_{78p}	$m'_{8,10,p}$	$\beta'_{8,10,p}$	$\mu'_{8,10,p}$	$\rho'_{8,10,p}$	$m'_{jk'p}$	$\beta'_{jk'p}$	$\mu'_{jk'p}$	$\rho'_{jk'p}$				
1	1	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0
2	1	0	0	0	2	0.25	1	0	1	0	0	0	0	0	0	0
3	1	0	0	0	3	5/12	2	0	1	0	0	0	0	0	0	0
4	2	1/3	1	0	1	0	0	0	1	0	0	0	1	0	3200	0
5	2	1/3	1	0	2	0.25	1	0	1	0	0	0	1	0	3200	0
6	2	1/3	1	0	3	5/12	2	0	1	0	0	0	1	0	3200	0
7	3	2/3	2	1	1	0	0	0	1	0	0	0	2	1	6400	3100
8	3	2/3	2	1	2	0.25	1	0	1	0	0	0	2	1	6400	3100
9	3	2/3	2	1	3	5/12	2	0	1	0	0	0	2	1	6400	3100

4.2 Project gain maximization and makespan minimization problems

The illustrative case was implemented in AMPL Studio v1.6.j and solved with Cplex 12.2. The gain maximization problem and the makespan minimization problem are investigated in this section. The latter is formulated without any upper bound for the overlapping cost. For each of this optimization criterion, the scheduling problems with or without resource constraints and with or without overlapping modes are presented.

Table 7 highlights that a significant reduction of the optimal makespan is obtained with overlapping : 11.11% for the scheduling problem with resource constraints and 15.53% for the scheduling problem without resource constraints. For the scheduling problem without resource constraints and with overlapping modes (case 3), almost all overlappable activities are overlapped (13 out of 15) and most of the couples of overlapped activities are overlapped at their local minimum (10 out of 13). In addition, Table 8 shows that all overlappable activities on the critical path obtained without overlapping modes (case 1) are overlapped in the scheduling problem with overlapping modes, as any reduction of the time to execute critical activities will decrease the project makespan. The project gain obtained with the makespan minimization problem with resource constraints is slightly positive. By contrast, the optimal makespan and the corresponding gain found in cases 1 and 3 show that minimizing the makespan using overlapping strategies can result in a lower gain, because the overlapping costs may exceed the opportunity cost.

Table 7 Effects of resource constraints and overlapping on the optimal project makespan

Case	Resource constraints	Overlapping modes	Number of overlappable activities	Number of overlapped activities	Optimal makespan	Corresponding gain	CPU's Time (s)
1	No	No	0	0	103	25000	0.09
2	Yes	No	0	0	108	0	0.36
3	No	Yes	15	13	87	23000	2.09
4	Yes	Yes	15	7	96	3000	1708.05

Table 8 Critical activities with and without resource constraints

	Case 1	Case 3
Critical activities	(1, 2, 3, 7, 8, 10, 13, 21, 23, 24, 25, 31, 32)	(1, 2, 3, 4, 5, 6, 7, 8, 10, 12, 13, 18, 20, 21, 23, 24 ,25 ,31,32)

When resource constraints are considered, overlapping is less performed than without resource constraints. As expected, overlapping lead to additional workload and to more resource consumptions. Overlapping is thus less attractive and less than half of the set of overlappable activities are overlapped with resource constraints and overlapping modes (case 4). This confirms that overlapping and resource constraints are closely interrelated. The overlapping modes of the optimal makespan obtained for both the makespan minimization and the gain maximization problems are detailed in Table 9.

Table 9 Effects of resource constraints and overlapping on the optimal project gain

couple of overlappable activities	Makespan minimization problem		Gain maximization problem	
	Case 3	Case 4	Case 3	Case 4
	overlapping mode	overlapping mode	overlapping mode	overlapping mode
(2, 3)	**3**	**3**	**3**	**2**
(3, 6)	**2**	**2**	**2**	1
(6, 18)	**2**	**2**	**2**	1
(7, 8)	**3**	**3**	**2**	1
(8, 10)	**3**	**3**	**2**	**2**
(10, 12)	1	1	1	1
(10, 13)	**3**	**3**	**2**	**2**
(11, 14)	**2**	**2**	**2**	1
(11, 17)	1	1	1	1
(13, 21)	**2**	**2**	**2**	1
(19, 29)	**3**	**3**	1	1
(20, 21)	**2**	**2**	1	1
(24, 25)	**3**	**3**	1	1
(25, 31)	**2**	**2**	**2**	1
(28,30)	**3**	**3**	1	**3**

The results obtained for the project gain maximization problem in Table 10 show that the optimal gains in cases 3 and 4 (with overlapping modes) are significantly better than those cases 1 and 2 (without overlapping), and better than the gain obtained for the project makespan minimization. However, the corresponding makespan is not as good as for the project makespan minimization. Indeed, the optimal schedules in terms of project gain lead to a 8.33% and a 12.50% reduction with and without resource constraints, respectively. Tables 7 and 10 also reveal that overlapping modes significantly increase the computational time required to solve the optimization problems, as it adds further complexity to the already complex case of resource-constrained scheduling problem, which is known to be a NP-hard optimization problem (Herroelen 2005).

Table 10 Effects of resource constraints and overlapping on the optimal project gain

Case	Resource constraints	Overlapping modes	Number of overlappable activities	Number of overlapped activities	Optimal makespan	Corresponding gain	CPU's Time (s)
1	No	No	0	0	25000	103	0.09
2	Yes	No	0	0	0	108	0.36
3	No	Yes	15	9	62400	91	2.09
4	Yes	Yes	15	4	38200	99	1251.41

In conclusion, the optimal schedules obtained for the gain maximization and the makespan minimization problems differ in terms of resulting gain and makespan. Minimizing the project makespan requires overlapping several couples of activities, which entails additional rework and communication/coordination costs. In order to conciliate these two contradictory targets, we propose to investigate the time-cost trade-off in the next section.

4.3 Time-cost trade-off analysis

This section aims to provide a deeper insight into the time-cost trade-off involving the opportunity cost for finishing the project earlier or later and the overlapping costs, which is composed of the rework costs and the communication/coordination costs. We also investigate the effects of the resource constraints on the time-cost trade-off by varying the resource availabilities to upper and lower values. Figure 3 presents the effects of overlapping on the project performances. These results were derived with the project gain maximization programming model described in section 3.1. An additional constraint on the project makespan was introduced in order to compute the project gain maximization problem for different values of the project makespan. The corresponding overlapping costs are depicted in Figure 4. For each resource availability, the range of tested values for the project makespan is bounded by the optimal makespans obtained with the makespan minimization problem with overlapping modes (lower bound) and without overlapping modes (upper bound), respectively.

Figure 4 highlights that the cost of decreasing the project makespan by one unit time increases as the project makespan decreases. Indeed, the overlapping data generated in the numerical example are such that the rework cost and duration, as well as the communication/coordination cost and duration are non-decreasing with respect to the amount of overlap for each couple of overlappable activities (see Table 5). In addition, more activities are executed in parallel and more and more couples of overlappable activities must be overlapped to gain one unit time when the project schedule is compressed. As shown in the previous section, the scarcer the resources, the more restricted the set of potential overlapped activities required to reduce the project makespan by one unit. Consequently, it becomes more expensive to overlap and reduce the project makespan when the resource availability is lower.

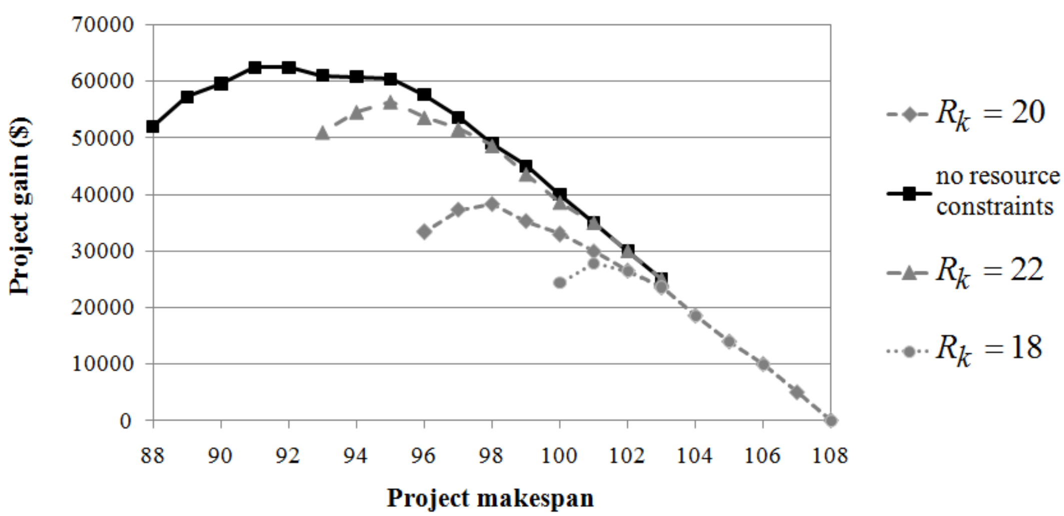

Fig. 3 The effects of overlapping on the project performances

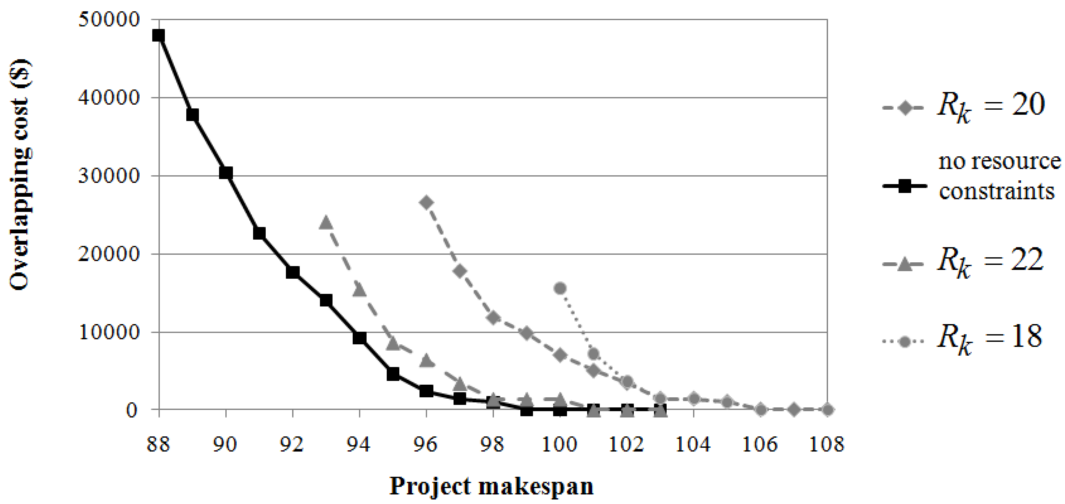

Fig. 4 Minimum overlapping cost as a function of the project makespan

As the overlapping costs required to reduce the project makespan by one unit time increase, it may outweigh the opportunity cost. Figure 3 shows that project gain decreases when activities are overlapped beyond a certain point. This explains why the optimal makespan obtained with the project gain maximization problem in the previous section differs from the minimal makespan. Depending on the resource availability and how much the project managers and planners are willing to pay to complete the project earlier, the time-cost trade-off analysis presented in this section should help them to choose an appropriate overlapping strategy.

5 Conclusion and Discussion

Overlapping activities is one of the most applied strategies to accelerate a project either in its early stage when the schedule baseline is set up or following project delay during its execution. Overlapping entails that downstream activities start before the information they require is available in a finalized form. However, the additional communication and coordination and the additional workload required to accommodate the information changes transmitted by upstream activities to the overlapped downstream activities are often ignored in practice. Moreover, in spite of all research efforts accomplished in evaluating the relation between the amount of overlap and rework and determining the optimal overlapping strategy for two activities without resource constraints (Krishnan et al. 1997; Loch and Terwiesch 1998; Roemer et al. 2000; Lin et al. 2009; Lin et al. 2010), only few papers have incorporated overlapping in the RCPSP of whole projects (Cho and Eppinger 2005; Gerk et al. 2008). In addition, these papers studied simplified linear rework model that are not realistic and considered instantaneous and costless information exchange.

We investigate the joint optimization of overlapping and resource-constrained project scheduling problem with the following assumptions: (1) preliminary information can be exchanged between identified overlappable activities, (2) the information flow is unidirectional from upstream to downstream activities, (3) information exchange require non-negligible time and cost, (4) overlapping is restricted to a finite number of feasible amounts of overlap for each couple of activities, corresponding to overlapping modes, and (5) the rework and the communication and coordination policy are preliminary estimated for each overlapping mode. The main contribution of this paper is to present a linear integer programming model for the project gain maximization problem and the project makespan minimization problem with overlapping modes and resource constraints. The objective is to find the project schedule and the overlapping levels of all couple of overlappable activities so as to maximize the project gain or minimize the project makespan.

An illustrative example provides several important findings. As overlapping entails additional workload and more resource consumptions, the resource availability limits the potential benefits of overlapping. Even without resource constraints, the overlapping strategy that consists in complete overlapping for all couples of overlappable activities (or overlapping at the maximum feasible amount of overlap), and the overlapping strategy that consists in overlapping each couple of overlappable activities at its local minimum are not optimal in terms of project gain and project makespan. Finally, minimizing the project makespan requires overlapping several couples of activities, which entails additional rework and communication/coordination costs. Therefore, maximizing the project gain and minimizing the project makespan are two contradictory objectives, and a time-cost trade-off analysis should assist project managers and planners to choose the most suitable overlapping strategy with respect to their cost and time objective.

Nonetheless, we would like to point out several limitations of our approach, and suggest some possible directions for future research. First, the proposed problem formulation with overlapping modes shares similarities with the traditional multi-mode resource constraint scheduling problem (MRCPSP). Considering the limit of exact solution procedure encountered with MRCSPSP, we can anticipate that solving the RCPSP with overlapping modes for larger projects, as they usually appears in practical cases, will require the use of metaheuristics or heuristics. Second, the relaxation of the aforementioned assumptions, such that the assumption that information flow is unidirectional, also represents interesting perspectives. Third, it is important to test the model with many instances in order to generalize our conclusion and to capture the effects of different parameters, such that the number of overlappable activities, the number of precedence relations, the resource consumption and the resource availability. Fourth, even though the estimation of the overlapping rework and the required communication and coordination policy for each couple of overlappable activities can be derived from historical data when the organization has experience with similar projects, the problem of how to reliably estimate these data for projects with which the organization has less familiarity should also be investigated. Finally, the problem is formulated in a deterministic environment and does not directly address schedule risks. However, overlapping is inherently risky as it entails that downstream activities start before the information they require is available in a finalized form. We may extend the model to introduce randomness along several parameters, such as the activity duration, the rework duration and the communication/coordination duration.

References

Blackburn JD, Hoedemaker G, Van Wassenhove LN (1996) Concurrent software engineering: Prospects and pitfalls. IEEE Transactions on engineering management 43:179–188

Browning TR (2001) Applying the design structure matrix to system decomposition and integration problems: a review and new directions. IEEE Transactions on engineering management 48(3):292–306

Browning TR, Eppinger SD (2002) Modeling impacts of process architecture on cost and schedule risk in product development. IEEE Transactions on engineering management 49(4):428–442

Cho SH, Eppinger SD (2005) A simulation-based process model for managing complex design projects. IEEE Transactions on engineering management 52(3): 316–328

Clark KB, Fujimoto T (1991) Product development performance strategy, organization and management in the world auto industry. Harvard Business School Press, Boston, MA

Dzeng RJ (2006) Identifying a design management package to support concurrent design in building wafer fabrication facilities. Journal of construction engineering and management 132(6): 606–614

Gerk JEV, Qassim RY (2008) Project acceleration via activity crashing, overlapping, and substitution. IEEE Transactions on Engineering Management 55(4):590–601

Hartmann S (1999) Project scheduling under limited resources. Springer, Berlin

Hartmann S, Briskorn D (2010) A survey of variants and extensions of the resource-constrained project scheduling problem. European Journal of Operational Research 207(1):1–14

Kolisch R, Sprecher A (1996) PSPLIB – A project scheduling problem library. European Journal of Operational Research 96(1):205–216

Krishnan V, Eppinger SD, Whitney DE (1997) A model-based framework to overlap product development activities. Management Science 43(4):437–451

Lin J, Chai KH, Brombacher AC, Wong YS (2009) Optimal overlapping and functional interaction in product development. European Journal of Operational Research 196(3): 1158–1169

Lin J, Qian Y, Cui W, Miao Z (2010) Overlapping and communication policies in product development. European Journal of Operational Research 201(3):737–750

Loch CH, Terwiesch C (1998) Communication and uncertainty in concurrent engineering. Management Science 44(8):1032–1048

Pena-Mora F, Li M, (2001) Dynamic planning and control methodology for design/build fast-track construction projects. Journal of Construction Engineering and Management 127(1): 1–17

Roemer TA, Ahmadi R, Wang RH (2000) Time-cost trade-offs in overlapped product development. Operations Research 48(6):858–865

Roemer TA, Ahmadi R (2004) Concurrent crashing and overlapping in product development. Operations Research 52(4): 606–622

Sabbagh K (1996) Twenty-first century jet. Scribner, New York

Smith PG, Reinertsen, DG (1998) Developing Products in Half the Time, 2nd edn. John Wiley, New York

Terwiesch C, Loch CH (1999) Measuring the effectiveness of overlapping development activities. Management Science 45(4): 455–465

Wang JT, Lin YI (2009) An overlapping process model to assess schedule risk for new product development. Computers & Industrial Engineering 57(2):460–474