Solving Planning and Scheduling Problems in Network based Operations

Christophe Guettier

SAGEM Defense and Security

Abstract. Future Network based Operations (NbO) will necessitate new approach to deal with assymetric context, urban area and tactical mobility. These operations will have to cope with multiple collaborative actions in a complex and changing environment. In NbO, collaborative actions stresses coordination and synchronisation requirements, both in terms of complexity and tempo. Consequently, NbO deeply impacts plan elaboration and task scheduling processes performed by the different units. This paper presents a constraint-based method for automating mission planning and scheduling in the context of Network based Operations. Relying on graphs, flow and timing formulations, constraints are easily formulated, comprising logical predicates which model coordinations and synchronisations. Search techniques, which combine variable ordering and concurrent solving, are also presented and evaluated on real world scenarii. Results show that mission planning and scheduling tools can be used to develop, experiment and evaluate the NbO concept.

1 Introduction

The new defence concept Network based Operation (NbO)[CICDE 2006] enables decision superiority thanks to a better information sharing between commanders. In classical operations, Planning and Scheduling (P&S) have been so far considered as separated off-line activities with limited automation, in particular at the tactical levels. In NbO, P&S plays a critical role still in terms of mission preparation, but also to support execution, and systems interoperability down to the tactical level. In modern operations taking place in urban environment, this level is subject to unstructured threats, versatile oponents and requires high tactical mobility. To be efficient and deliver the expected effects, NbO necessitate strong coordinations and synchronisations between the different units. This is emphasized in urban areas and peace keeping missions, which characterise most of modern operations. Consistency between plans and schedules becomes critical to insure coordination of actions and to deliver the required effects. Therefore, not only automation is strongly recommanded, but P&S must be tackled as a global and composite problem.

This paper focuses on P&S problems for the tactical level, in a NbO context. A complete constraint-based approach is proposed, including problem formulation, modelling and search design. Experimentation exhibits interesting performances to evaluate the NbO concepts in multiple conditions (urban vs open environment, peace keeping vs high intensity). A P&S tool is presented, which can be integrated either in a future battle management system or within a defence laboratory in order to apply on real world scenarii.

Mainly in space and defence areas, mission planning problems have always been a major challenge for the planning community, in terms of problem formulation, modelling, search techniques and evaluation. Generic formalisms have highlighted problem complexity and some domain independent search and heuristics proposed. However, most of the approaches considering realistic operational requirements have been so far dedicated to the problem. To this end, constraint-based techniques have already been combined with planning formalisms in [Muscettola 1998] [Allo & al. 2001]. Complex task scheduling has been a major area of investigation within the constraint programming community. The expressive power of constraints through logical connectors and quantifiers, combined with powerfull solving methods enables one to address wide, large-scale classes of combinatorial problems. Moreover, constraint programming has been able to successfully integrate results in Operation Research (OR) and Linear Programming (LP).

Following this approach, a flow-based $\{0,1\}$ modelling of planning problems is proposed, steming from the state of the art in Operation Research (OR). This offers a straightforward and natural expression of constrained path planning, also supporting multiple flows and various weighting metrics. The formulation uses a graph representation of the terrain and is very practical to represent tactical mobility (positions, progression axis, objectives), unit movements and to locate actions in space and time. Metrics are used to represent distances, protection, or resource consumption during certain movements or actions. This modelling approach also provides closed forms of transitions between actions, which facilitate the specification of coordination predicates and scheduling constraints. Different search techniques are evaluated to solve the P&S problem. They all use branch and bound optimisation techniques, combined with constraint propagation algorithms. Advanced search methods have been designed using different combination of heuristics, concurrent solving and variable ordering. Experimentations on realistic examples illustrate the efficiency of the approach. The P&S tool is deployed in a battle lab to support the development, experimentations and evaluations of future NbO concepts at the tactical level. In particular, the tool has been shown very useful to understand how far digital information can optimise operation management.

The paper is organised as follow: the problem is presented in section § 2, a CP approach is motivated in § 3 and detailed in § 4.

2 Problem

This section details the context of Network based Operation (§ 2.1), followed by example (§ 2.2) illustrating a first informal P&S problem presentation (§ 2.3).

2.1 Context

In NbO, the high quality of information sharing between commanders enables a better understanding of opponent intent as well as a wider scope of options to consider [CICDE 2006]. All parties involve in a given military action shall have the same understanding of the situation at all times, a clear idea of available resources to engage, and real-time coordination capabilities between the different units. NbO is a promising approach in terms of operational capabilities and optimisation (accelerating the peace of events, optimising resources, minimizing frictions, . . .). Exploiting NbO concept, it is possible to anticipate the ennemy course of action and to decide faster using the relevant information. This concept is related to Network Centric Warfare (NCW)[Alberts & al 2001] where ”information advantage enables decision superiority”. In NCW, any subset of the army components can synchronise their action at any time in order to deliver the required effect. In contrast with NbO, action planning and task scheduling are controled from the top hierarchy down to elementary units. In Network Enabled Capabilities (NEC), synchronisations between the actions of different army components are planned according to the effects. NEC is very close to NbO and requires a fine interleaving of off-line deliberate planning, ”just-in-time” replanning and tasking. All these approaches rely on the so-called Common Operational Picture (COP), composed with situation reports indicating geographical location, position and situation of allies, neutral and hostile units.

But P&S is becoming a core issue, in particular for the tactical level, where limited time is available to make decisions [Hayes & al 2005]. These units execute their mission with timing pressure, limited resources, mobile oponents with unstructured threats. As a matter of fact, this is emphasized in most of modern and peace keeping operations, taking place in urban environment under assymetric engagement conditions (where no clear front line exists). P&S occur at preparation time but is also necessary to manage dynamically the mission, when contingent events happen (system failures, incidents, waste of time or ressources, . . .).

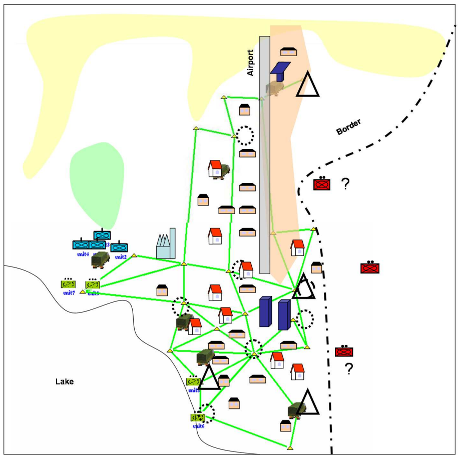
These difficulties can rapidly turn into vulnerabilities without a high degree of tactical mobility, coordination and synchronisation [Houghton 2004]. Assuming an available COP, the problem is to construct a course of actions with a consistent timeline in order to meet mission objectives. This problem is constrained by coordination requirements, available resources as well as terrain invariants (progression axis, positions, distances and protection). Coordination is strongly structuring the problem and results from the expected effects to be achieved (for example surprise, exploiting opponent vulnerabilities, . . .), collaborative actions (such as collective observations, collaborative protections, . . .) and interoperability constraints (for instance, the fact that two systems cannot be used simultaneously in a same area).

2.2 Example

As an illustrative example, let us consider the following situation inspired from a real case. United Nations (UN) peace keeping forces are deployed in the town depicted in Fig. 1 close to the eastern border with an unstable country. Coming from the west, a reinforcement mission comprising eight tactical units consists in securing UN locations, hospitals, airport and civil administrations, in spite of hostile intrusions from the east.

However, actions consumes unit capabilities (energy, water, tiredness). Also, these actions need to be coordinated and the following rules of engagement (RoE) must be applied:

- in urban areas, any protection action must be preceeded by a reconnaissance one,
- two units cannot manoeuvre simultaneously on a given town position or location,



The COP given at the start of the mission is represented by blue (left) and red rectangles (right) for friendly and hostile forces respectively, and locations of mission primary and secondary objectives are figured out by triangles and dotted circles, respectively. Units can progress along main streets to reach UN positions (or primary objective) and must secure few urban areas (or secondary objectives). Units must realise some reconnaissance, observation and protection actions at location of primary and secondary objectives. Green lines show possible progression axis between main tactical positions downtown. Other actions must also be realised depending on the intermediate positions situated on a unit path. Units must also satisfy coordination constraints, for example preventing from simultaneous manoeuvre at the same cross-street.

Fig. 1. *COP example giving the situation at mission preparation time.*

- any action on main town cross streets (or round-about) must find some support from a different unit, recently deployed at an adjacent position.
- primary objectives are reached simultaneously.

These sorts of coordination constraints make the progression plan complex to elaborate. Evenmore, a schedule of events must be found in order to evaluate mission feasibility. Lastly, the reinfocerment units must be deployed onto primary objectives in minimal time. Of course, when unexpected threats or contingent events occur (in the example, it may happen from the border), the COP is updated and both plan and schedule must be adapted.

2.3 Formulation

A simplified and informal input specification can be expressed using terrain structure, initial conditions, mission objectives, unit capabilites and coordination constraints. The following elements are known off-line and characterise this input specification:

- Terrain structure: is defined as a set of positions, related by progression axis. Each position has a geographical location and two adjacent positions are separated by a given distance.
- Initial conditions: are the resources initially available per unit, the initial positions of friendly and hostile units.
- Objectives: Some of the positions can correspond to secondary or primary objectives, with mandatory actions to realise.
- Unit capabilities: are formulated using mobility constraints (the minimum and maximum possible speeds on a progression axis), actions that can be realised on a given position and lastly resources required and consumed by a given action. Also, some positions or progression axis are more or less dangerous according to the unit.

– Coordination constraints, resulting from expected effects and RoE: impose synchronisations between units whenever it is required. This can constrain unit location and timeline but also impose units to realise actions in parallel.

The terrain is represented as a directed graph structure (see Fig. 1), where edges define progression axis and vertices tactical positions (or locations). Graph structure is a convenient way to take in account real world details or to scale the problem difficulty where it is necessary. Vertices also represent primary and secondary objectives. Each action is represented by a position, a realisation time, a resource (or capacity) consumption, and a required amout of initial capacity. Other constraints can impose vertices or edges to be excluded or included in a given unit plan. Lastly, protection and capacity metrics are associated to the edges, for each unit.

The problem is to find, for each unit, a sequence of actions and movements (e.g. a plan) with an associated timeline from the initial position to the primary objective. When a movement among a progression axis belongs to the plan, its speed must also be found. The plan must include all secondary objectives as well as satisfying coordination constraints. Three constraints can be considered between any couple of units:

– Support: Unit A must perform a synchronised action on a position X, if unit B reaches a given position X'.
– Composite actions: When two units perform a "composite" action (or couple of actions) on two positions, they have to be synchronised.
– Exclusive actions: When two units reach a given couple of positions, they must not perform actions simultaneously.

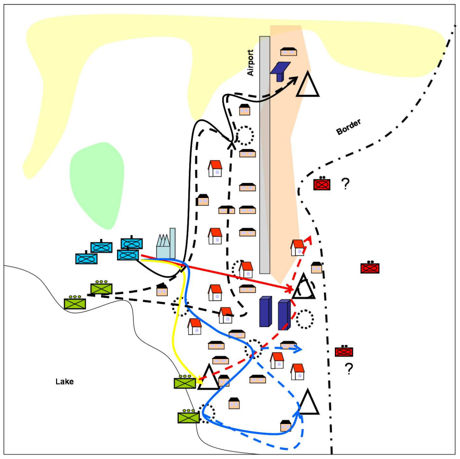
The solution (see Fig. 2) can also be assimilated as a path for each unit, which is represented by a succession of movement along progression axis, and where some actions have to be realised on the different positions. Note that in contrast with classical task scheduling problems, all possible actions may not be realised. For each unit, the set of actions to be realised depends on its path, which in turn is constrained by other unit plans.

3 Why CP?

The synthesis of existing practices (§ 3.1) and related work (§ 3.2) is given. This also motivates the CP approach (§ 3.3) for tackling the P&S problem.

3.1 Domain practices

A lot of researches, experiments and development have been carried forward in NbO, NEC or NCW, but rather focusing on air and naval operations, at strategical levels. The US Future Combat Systems focuses on land capabilities at the tactical level, but with rather symetric engagement conditions. To our knowledge, there is no operational automated P&S tool focusing on the tactical level, and able to consider both urban environment and assymetric conditions. Compared to P&S for air or naval operations, the requirements are less time-critical, but the level of difficulty is more important,



On the example, primary mission objectives are located near the border (such as airport, hospital, main cross-street). Objectives are reached simultaneously to produce a coordinated effect, compatible with global movement pace. A path is assigned to each unit, dotted and plain lines correspond respectively to reconnaissance and protection unit plans. Different positions have to be cleared to perform this mission, for which many coordination constraints have been defined. In particular, most of key positions have to be left by reconnaissance unts before protection units arrive. This can be achieved by synchronising progression of reconnaissance (dotted lines) with protection ones (solid lines).

Fig. 2. *Plan solution to reinforce UN peace keeping forces*

due to the environment complexity and the number of heterogeneous parameters and constraints to consider simultaneously.

Two different practices are presented to illustrate current methods. Resulting from its doctrine, US Army has a top-down approach of P&S. The high level command can decide precisely all actions and their rough timeline down to the squad (or even team) level (which is very low-level tactic). Low level commands have few degree of freedom on the plan and report on task execution. This centralised approach makes P&S difficult, since the whole problem must be solved in a single process. At the opposite, the French army doctrine is basing P&S decisions on the subsidiarity principle. This means that a given command level gives objectives to the immediate lower level, abstracting away other lower level actions. These objectives result from a local level planning method, which tend to relax scheduling constraints in order to allow maximal flexibility and reactivity to lower levels. In turn, with this decentralised approach, a global and full detailed plan is not elaborated since a given level reports mainly on their objectives.

The following table gives a *rough* order of magnitude in terms of number of units and time to take decisions.[1] In both cases, P&S automation is required to facilitate decision making and to relief commanders. P&S is also a mandatory service to achieve NCW, NEC or NBO concepts.

	team/squad	company/squadron	battalion	brigade	division
time for P&S decisions	second to 5 mins	5 to 15 mins	hour	1 to 6 hours	6 to 8 hours
number of units below	4	16	70	300	1500

[1] The battalion level combines multiple arms of land forces (artillery, cavalry, infantry, . . .), while brigade level or ”joint” is the upper tactical level, that also includes air and naval armies. Indeed, real data depends on the country and the mission.

3.2 Related work

Generic P&S: Generic planning under resource management is also a related field of research. According to [Long and Fox 2000], the problem belongs to transportation problem classes and a set of pre-planning algorithms can identify the problem class. By knowing the problem class, it is possible to select automatically the right set of dedicated heuristics. The latest version of the generic planning description language PDDL [Ghallab et al. 1998] is extended to consider resource management and timing issues. Planning techniques have been applied to analyse plans and capabilities in many defence applications [Myers 2002] [Myers 2006].

Domain specific P&S: Much planning work has been led in the US for both military or civilian purposes, using dynamic programming. Many approaches rely on Hierarchical Task Network (HTN). Example can be found in MACBeth (see [Goldman et al. 2000]), which develops a large set of possibilities to specify a problem constrained in time and resource. MacBeth is domain independent and can be applied to air or land missions. This planner combines HTN with constraint propagation techniques to prune the search space. The MISURE project also exhibits a constraint-based planning technique, but tuned to air operations [Allo & al. 2001].

Constraint-based P&S: Few approaches are currently used for P&S with resource constraints. The first uses data structures to model the resources, and apply the solving process to the problem composed of actions and resources. This is the case in IxTeT [Laborie and Ghallab 1995] and HSTS [Muscettola 1994], for example. LPSAT uses floats and fluents to model constants and variables and couples a SAT solver with a simplex method to manage both actions and resources. A second approach consists in selecting the actions by solving a first planning problem and then consider the resource management as a second scheduling problem. This is the case in parcPlan [Lever and Richards 1994] and RealPlan [Srivastava 2000]. Lastly, extensions of Concurrent Constraint (CC) [Saraswat et al. 1993] language have been the focus of researches led in NASA and MIT. In their Reactive Model-based Programming Language (RMPL) [Kim & al 2001], an evolution of CC languages, the same paradigm is used to dynamically constrain planning representations of one or more remote agents. RMPL applications include multi-robot coordination, mission execution management for aircraft fighters.

3.3 CP approach

Operational users are not only interested in mission feasibility, but also in its optimisation. Resulting from user experience, two kinds of optimisation are interesting, minimizing mission duration and maximizing mission safety. It is also important to express domain heuristics and to rapidly adapt the P&S tools according to environment, terrain and missions. This can be achieved using CP expressiveness, which under a model-based development approach, also enable the management of tool evolutions.

A CP approach is interesting in many respects. The problem is global, composite and requires the formulation of different related models. This can be achieved in a natural way with logical constraint composition using classical operators [Van Hentenryck et al. 1995]. This is more particularly the case for modelling coordination, which involves disjunctive constraints and temporal predicates. By introducing these constraints relating different paths, it becomes very difficult to take advantage of OR techniques based on path algebra, multicommodity flows or linear programming. Similarly, LP techniques will have to cope with non-linear constraints and discrete variables which cannot be easily recasted into linear ones without a massive increase of the variable set. However, most of constraint programming framework are usefull to design hybrid search techniques, by integrating OR and LP algorithms [Ajili & Wallace 2003]. This method is followed to solve the P&S problem by exploiting Dijkstra algorithm to elaborate a meta-metric on search exploration. This work can be easily extended to perform dynamic probing. Same framework also enable concurrent solving, which is also a serious option to design search strategies and can be combined with hybrid search.

3.4 Discussion

The approach does not consider replanning techniques, as a focus on solving the global problem is addressed first. Of course, this is highly relevant for operational users and the approach can be easily extended to develop such techniques. Likewise, contingent planning is not considered. In contrast, introducing contingency formulations can significantly impact the model proposed in this paper.

4 How CP?

The P&S constraint-based models are defined in (§ 4.1) and runs an optimisation strategy (§ 4.2). Examples of experiments are proposed in § 4.3.

4.1 Modelling

The space of possible plans is represented as a directed graph $G(X, U)$ where the set of edges U is representing possible progression axis and the set of vertices X possible position (or navigation) locations.[2] In the following, we consider n units and $k,\ k \in \{0,\ \ldots n-1\}$ denotes a given unit. A unit starts from vertex $start_k$ and must reach its objective at vertex end_k.

Progression axis and mobility actions A path of progression axis is defined by the set of positive flows. For each unit, a set of variables $\varphi_u^k \in \{0, 1\}$ models a possible path from $start_k$ to end_k, where the edge u belongs to the path of unit k if and only if decision variable φ_u^k is instantiated to 1. When a progression axis is selected as part of the unit path (e.g. $\varphi_u^k = 1$), it can also be assimilated as an elementary mobility action by unit k on edge (or progression axis) u. Mobility actions, a strong part of the mission plan, can be represented as $\Phi_k = \{u|\ u \in U,\ \varphi_u^k = 1\}$ for unit k (see Fig. 3).

[2] In the remaining of the paper, a vertex is denoted x, while an edge can be denoted either u or (x, x').

Path consistency: From an initial position to a final one, path consistency is asserted by the following constraints, where $\omega^+(x) \subset U$ and $\omega^-(x) \subset U$ are outgoing and incoming edges from vertex x, respectively.

$$\forall k \in 0 \dots n-1, \sum_{u \,\in\, \omega^+(start^k)} \varphi_u^k \;=\; 1, \sum_{u \,\in\, \omega^-(end^k)} \varphi_u^k \;=\; 1, \tag{1}$$

$$\forall x \in X \setminus \{start^k, end^k\}, \sum_{u \,\in\, \omega^+(x)} \varphi_u^k = \sum_{u \,\in\, \omega^-(x)} \varphi_u^k \leq 1 \tag{2}$$

Nodes $start^k$ and end^k represent respectively current position and primary objective of unit k. Equation (2) ensures path connectivity and unicity while equation (1) imposes limit conditions for the extremities of the path. This constraint gives a linear chain alternating positions and mobility actions (along progression axis) along the graph.

Path length and schedule formulations: For a given unit k, this formulation binds mobility actions Φ^k and mission schedule. Assuming a given date D_x^k associated with a position (e.g. vertex) x and a single unit k, we use a well known path length formulation (3) oftenly considered in OR [Gondran and Minoux 1995]. Variable D_x^k is expressing the time at which unit k reaches position x (see example in figure 3). Assuming that variable $d_{(x',x)}^k$ represents the time taken to perform the mobility action from position x' to x (or progression speed), we have:

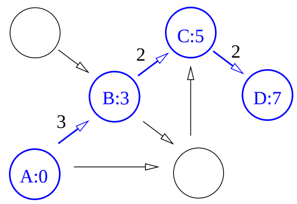
$$\begin{aligned} &\forall x \in X, D_x^k = \sum_{(x',x) \,\in\, \omega^-(x)} \varphi_{(x',x)}^k (d_{(x',x)}^k + D_{x'}^k) \\ &\forall (x,x') \in U,\ d_{(x,x')}^k \in \mathbb{N},\ l_{(x,x')}^k \leq d_{(x,x')}^k \leq u_{(x,x')}^k \end{aligned} \tag{3}$$

Note that upper and lower limits (resp. $u_{(x,x')}^k$ and $l_{(x,x')}^k$) in (3) are specified for a couple of unit and edge in order to take in account specific unit flexibility on a given progression axis. Indeed, variables $d_{(x,x')}^k$ are critical decision variables in the problem and make constraints (3) non linear. Finally, the mission schedule can be represented as $\Delta^k = \{(x, D_x^k) |\ x \in X,\ D_x^k > 0\}$. Note that a similar constraint-based formulation is also used for other mission metrics (Fig. 3), such as resource capacity.

Positions and static actions: The set of positions (or navigation points) T^k belonging to a given unit path k can also be expressed as follow (4):

$$\forall x, t_x^k = min(1, D^k(x)),\ T^k = \{D^k(x) |\ x \in X,\ t_x^k = 1\} \tag{4}$$

where t_x^k states whether a position x is part of the planned path for unit k. In the following logical formulations, t_x^k is assimilated as a boolean variable. It is possible to formulate a secondary objective to unit k on position x by imposing $t_x^k = 1$ off-line. The static action model is simpler. A possible static action for unit k on position x is



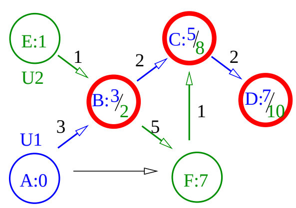
This graph is a spatial representation of progression axis (edges) and positions (nodes). Mobility actions, corresponding to the set of positive values $\Phi = \{(A,B),(B,C),(C,D)\}$, are represented with bold arrows. Assuming a timing metric (edge values are progression speeds), the schedule is $\Delta = \{(A,0),(B,3),(C,5),(D,7)\}$. Available energy, water, tiredness and security are similarly formulated in different experiments.

Fig. 3. *Illustrating mobility actions and schedule over a graph of positions and progression axis*

represented by $\langle t_x^k, \delta_x^k, \kappa_x^k \rangle$. Unit k must execute the action on position x iff $t_x^k = 1$. Integer constants δ_x^k and κ_x^k represent respectively action duration and capacity consumption. These constants extend respectively timing and capacity formulations from (3).[3]

Coordination constraints: In planning for multiple units acting collaboratively, it is necessary to define constraint schemes such as unit synchronizations, coordinations or composite actions. To formally define those constraints, we need to consider that plans can be related by causal dependencies and that they are conducted in parallel.

This consists in expressing constraints between pair of units k or k', involving the dates D_k^x or $D_{k'}^{x'}$ at which they reach positions x and x' (e.g. navigation points). Three kinds of constraints have been used to bind the set of mobility actions $\{\Phi^k\}_{k=0}^{n-1}$ with the set of associated schedules $\{\Delta_k\}_{k=0}^{n-1}$ (Fig. 4).



On the figure, U1 starts in A and U2 starts in E. Both units must reach D as final objective, and unit U1 must reach position C before U2 must leave position F. Still considering timing metrics (with progression speeds as edge values), a couple of satisfying schedules for U1 and U2 are respectvely $\Delta_1 = \{(A,0),(B,3),(C,5),(D,7)\}$ and $\Delta_2 = \{(E,1),(B,2),(F,7),(C,8),(D,10)\}$.

Fig. 4. *Illustrating mobility actions and schedules coordinations for 2 units*

– Support: When a unit k arriving to a given navigation point x, it needs synchronised support from a remote unit k' standing in navigation point x':

$$\forall k,\ support(k,k',x,x',c),\ t_k^x \ \Rightarrow\ t_{k'}^{x'} \ \wedge\ D_{x'}^{k'} \leq D_x^k + c \qquad (5)$$

where c is a constant.

– Composite actions: When two different units k and k' arrive to two navigation points x and x', a synchronisation is necessary:

[3] For simplicity (3) is not presented in its extended formulation.

$$\forall k,\ coordination(k,k',x,x',c)\ \wedge\ t_k^x\ \wedge\ t_{k'}^{x'}\ \Rightarrow\ D_{x'}^{k'} \leq D_x^k + c \qquad (6)$$

where c is a constant.
This is also considered as a ”coordinated” action, as unit k cannot act on navigation point x without another unit k' acting on a navigation point x' and vice versa.
– Exclusive actions: Two different units k and k' cannot arrive on two navigation point x and x' within a given time interval:

$$\forall k,\ exclusive(k,k',x,x',c)\, t_k^x \wedge t_{k'}^{x'}\ \Rightarrow\ (D_{x'}^{k'} \geq D_x^k + c) \vee (D_x^k \geq D_{x'}^{k'} + c) \quad (7)$$

In all these constraints, $c \in \mathbb{N}$. Note that the right hand side of the logical formulation involves a temporal predicates. Other temporal predicates can be used to model more complex coordination constraints. As a representative example, a disjuntive constraint can be used to guarantee that only one unit can be at a given navigation point in a time interval.

4.2 Solving Strategies

The solving strategies focus on mission duration optimisation, that is *minimising the maximal completion date*. This date correspond to one of the variable set $\{D_{end^k}^k\}_{k=0}^{n-1}$.[4] Designing the solving strategy consists in finding the right variables ordering and values filtering. Only complete searchs are considered in this work, and global optimality is also a challenge of interest for reasonable problem dimensions. All problem formulations and search strategies have been implemented in the $CLP(FD)$ SICStus prolog library. Basic solving techniques make use of branch-and-bound *minimize* predicate and $CLP(FD)$ constraint propagation algorithm.

On different realistic problem instances, any simple and naive strategy cannot find interesting solution in reasonable time. Consequently, three efficient solving strategies are proposed, giving feasible and opimised solutions relevant enough to operational users. On some problem instances, optimality can be achieved and proven by the strategy. For each unit k, decision variables are path variables $\{\varphi_x^k\}$, timing and duration variables, respectively $\{D_u^k\}$ and $\{d_u^k\}$.

– Feasible Path First (FPF): this basic strategy searches one by one unit path variables and then explore both speed and timing variables. Unit actions are automatically infered according to its path (sequence of progression axis and position, e.g. edges and vertices). Possible values of speed are enumerated using an increasing order.
– Directed Feasible Path First (DFPF): Instead of dynamic probing with tentative values [El Sakkout & Wallace 2000], this search strategy uses a static prober which build a quotient graph based on secondary objectives. Minimal path lengths are precomputed on quotient graph using Dijkstra algorithm to elaborate a distance metric. For each unit, path variables are statically ordered using that meta-metric. Then, the solving follows FPF strategy.

[4] The position end^k is the primary objective of unit k

- Directed Concurrent Path Solving (DCPS): The third strategy also makes use of the static prober to order path variables. However, it solves unit paths on a concurrent basis, using ”frozen goals” paradigms. In this technique, solving over a given subset of path variables is frozen, waiting for external coordination constraint to be entailed.

Discussion: Using the Dijkstra algorithm to solve a relaxed version of the problem highlights three important conclusions. It is possible to design efficient and simple hybrid algorithms using a CP framework, with important performance gains. This can be done with a static probing technique which enables the construction of meta-metrics, used for ordering problem variables. Lastly, variable ordering plays a critical role to solve efficiently the proposed problem instances.

4.3 Experimentations

Battle command and laboratory: Experimentations are performed in a battle laboratory, where all experts can use P&S to study real world scenarii. The P&S tool is part of a virtual battle command involving multiple users, which also interacts with different actors (simulated or real). Many scenarii can be studied faster than in the past, since users can rapidly specify capability metrics, objectives, actions and coordinations. Beyond mission feasibility and optimisation, users can also evaluate the impact of new capabilities (observation, mobility, effect,. . .) as well as specific mission characteristics, such as vulnerabilities, effects timeline and new command principles.

Benchmarks: To illustrate the approach, experiments on four benchmarks are presented. They are representative of modern peace keeping missions in the context of Network based Operations. All scenarii are relevant to the battalion level. Examples of units involved are special forces, recon and infantry companies, tank squadrons as well as helicopter and light artillery (Fig. 5).

1. Recon villages: deployment manoeuvre including several village reconnaissances, assimilated as secondary objectives. The tank squadron must support any infantery unit entering a village.
2. Reinforce UN positions: presented in (Fig. 1).
3. Sites inspections in urban area: several sites must be inspected in a town, and are formulated as secondary objectives. Town exits and entries must be secured during inspections. To keep the initiative, reconnaissance units must be previously deployed near the sites.
4. Secure humanitarian area: deployement manoeuvre for securing an area to gather refugees near a town. Secondary objectives are location of refugees. Protection forces must be coordinated to keep the surprise effect.

For each scenario, several instances are considered, corresponding to an increasing number of units $(2, 4, 6, 8)$ (Fig. 5). In all cases, for the purpose of this paper, only minimising mission duration is considered, although many other costs can be envisaged. Multiple metrics are also considered: progression duration, timing, protection and capacity consumption.

Results: Algorithms DFPF et DCPS behave in a similar way on all the benchmarks. Thus, only DFPF and FPF are compared in the following (see Fig. 5 and 6). DFPF proves optimality on 6 of the 16 instances, against 3 instances for FPF. In general, DFPF dominates FPF with a huge solving time ratio. Lastly, on the most difficult problem, *secure humanitarian area*, FPF is no longer competitive.

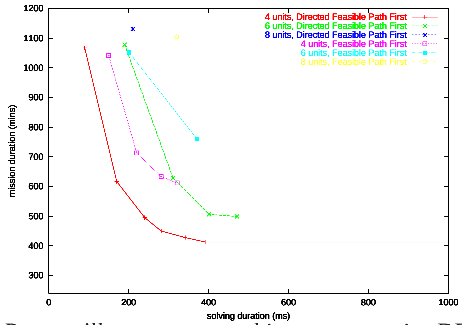
Problem characteristics						**FPF**			**DFPF**		
Units	Variables	Constraints	Actions	Coordinations	Second. objectives	Time (ms) for best sol.	Time (ms) for proving opt.	Best Value (minutes)	Time (ms) for best sol.	Time (ms) for proving opt.	Best Value (minutes)
1. Recon villages (22 nodes, 74 edges)											
2	1404	4913	3	0	4	1612	6412	397	141	2013	397
2	2808	9859	5	5	6	321		612	391	3866	413
2	4212	14757	7	7	7	371		760	471		499
2	5616	19664	9	10	8	320		1105	210		1131
2. Reinforce UN (23 nodes, 76 edges)											
2	1446	5129	2	4	2	15873	28111	109	30	2253	109
4	2892	10233	4	5	4	15942	176684	109	70	33338	109
6	4338	15333	6	7	5				130	32967	109
8	5784	20411	8	11	6	22483		161	140		161
3. Sites inspection in urban area (22 nodes, 68 edges)											
2	1308	4587	4	1	4	861		469	471		469
4	2616	9203	8	7	6	2433		520	270		520
6	3924	13802	10	10	8	2573		520	301		520
8	5232	18396	12	12	10				952		520
4. Secure humanitarian area (33 nodes 113 edges)											
2	2138	7612	3	5	1	105541		308	60		294
4	4276	15297	7	21	4				121		402
6	6414	22857	9	24	6				190		455
8	8552	30417	13	26	8				771		602

Fig. 5. *Results overview on benchmark scenarii, optimising mission time (in minutes)*

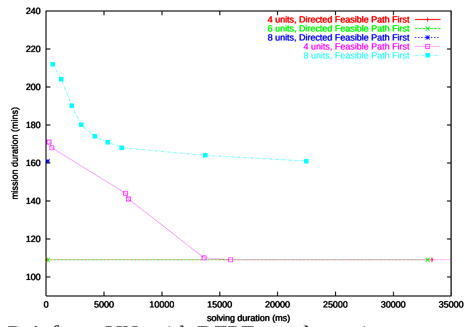
The two first benchmarks are the less constraining and are more optimisation problems (Fig. 6). At the opposite, the last benchmark is the most difficult as finding feasible plan and schedule is hard. Ratio between FPF and DFPF can be extreme, as for the 1^{st} instance of the 4^{th} scenario or the 2^{nd} instance of the 2^{nd} scenario. Note that important ratio can be also observed for solutions optimally proven. However, when values are found, they differ slightly only for the 1^{st} scenario. It is difficult to prove optimality for the two last scenarii, which are more constrained by secondary objectives, actions and coordinations.

5 Conclusion

A generic and full constraint-based approach has been proposed for solving P&S problems in the context of NbO. On the modelling side, CP models enable users to express



Recon villages: even on this easy scenario, DFPF gives better results and faster, except for the couple of runs with 8 units.



Reinforce UN: with DFPF, results on instances with 4,6 and 8 units are available in less than 200ms. FPF must execute many optimisation steps with 4 and 8 units.

Fig. 6. *Search behaviour (x and y axis are respectively solving time and mission duration).*

realistic P&S problems, while addressing timing, resource consumption as well as co-ordination of actions. In particular, combining path and coordination formulations is generic and powerful enough to formalise mission planning over several units. On the solving side, variable ordering is a critical aspect of optimisation performances. Efficiency of meta-metrics to construct a relevant variable ordering has been highlighted. Further work will focus on dynamic construction of meta-metrics using probing algorithms. The tool is beeing deployed in a battle lab, which increases users performances in analysing many realistic scenarii. Its integration in multiple sorts of future battle command is also seriously envisaged.

Acknowledgement

Col. Potel, Col. Hermellin and O.W. Berthaud H. must be acknowledged for their inputs and the quality of their explanations and advices. I also thank P. Rioufreyt, P. Curlier and J. Yelloz for their help and constant support. N. Beldiceanu and G. Verfaillie are also acknowledged for their constructive comments

References

[Ajili & Wallace 2003] Ajili,F.; Wallace,M. 2003. Constraint and Integer Programming: Toward a Unified Methodology. In Chapter 6: Hybrid Problem Solving in ECLiPSe. Kluwer Academic Publishers, 2003

[Alberts & al 2001] Alberts,D.; Garstka,J.; Hayes,R. and Signori D. 2001. Understanding Information Age Warfare, Library of Congress, USA, August 2001

[Allo & al. 2001] Allo,B.; Guettier,C.; Lecubin,N. 2001. A Demonstration of Dedicated Constraint-based Planning within Agent-based Architectures for Autonomous Aircraft. Proceedings of the IEEE Symposium on Intelligent Control.

[El Sakkout & Wallace 2000] El Sakkout,H. and Wallace M. Probe Backtrack Search for Minimal Perturbations in Dynamic Scheduling. Constraints Journal, Vol. 5, 2000.

[Ghallab et al. 1998] Ghallab,M.; Howe,A.; Knowblock,C.; Mac Dermott,D.; Ram,A.; Veloso,M.; Weld,D.; and Wilkins,D. 1998. PDDL- The Planning Domain Definition Language. Technical Report, CVC TR-98-003/DCS TR-1165.

[Gondran and Minoux 1995] Gondran,M. and Minoux,M.1995. *Graphes et Algorithmes*, ed. Eyrolles, Paris.

[Goldman et al. 2000] Goldman,R.P.; Haigh,K.Z.; Musliner,D.J.; and Pelican,M. 2000. MACBeth: A Multi-Agent Constraint-Based Planner. Proceedings of the AAAI Workshop on Constraints and AI Planning. Menlo Park, Calif.: AAAI Press.

[Hayes & al 2005] Hayes,C.; Larson, A.; Ravinder, U., 2005. Weasel: A mixed-Initiative System to Assist in Military Planning. Proceedings of the 15th Int. Conference on Automated Planning and Scheduling. 2005, USA.

[Houghton 2004] Houghton P., 2004. Potential Vulnerablities of a Network Enabled Force. Proceedings of ICCRT'04.

[CICDE 2006] Centre Interarme Concept Doctrine Exprimentation, Concept Exploratoire Oprations en Rseau. Report, 300-Def-CICDE-LIP, Oct. 2006.

[Laborie and Ghallab 1995] Laborie,P. and Ghallab,M. 1995. Planning with sharable resource constraints. Proceedings of IJCAI'95. Menlo Park, Calif.: International Joint Conference on Artificial Intelligence, Inc.

[Long and Fox 2000] Long, D. and Fox, M. 2000. Automatic Synthesis and use of Generic Types in Planning. Proceedings of AAAI 2000. Menlo Park, Calif.: AAAI Press.

[Lever and Richards 1994] Lever,J.M. and Richards,B. 1994. ParcPLAN: a planning architecture with parallel actions, resources and constraints. In Proceedings of the 9th International Symposium on Methodologies for Intelligent Systems.

[Muscettola 1994] Muscettola,N. 1994. Integrating Planning and Scheduling. In Intelligent Scheduling. Zweben and Fox, eds. Morgan-Kaufmann.

[Muscettola 1998] Muscettola,N.; Pandurang Nayak, P.; Bell,P. and Williams, B.C. Remote Agent: To boldly go where no AI system has gone before, NASA Ames Research Center. 1998.

[Myers 2002] Myers,K. Active Coordination of Distributed Human Planners, Proceedings of the 6th Int. Conference on Artificial Intelligence Planning and Scheduling. 2002.

[Myers 2006] Myers,K. Metatheoretic Plan Summarization and Comparison, Proceedings of the 16th Int. Conference on Automated Planning and Scheduling. 2006, UK.

[Srivastava 2000] Srivastava,B. 2000. RealPlan: Decoupling Causal and Resource Reasoning in Planning. Proceedings of AAAI 2000. Menlo Park, Calif.: AAAI Press.

[Saraswat et al. 1993] Saraswat,V.;Bobrow,D. and de Kleer,J. Infrastructure for Model-based computing TR, Xerox PARC, Palo Alto Ca.

[Van Hentenryck et al. 1995] Van Hentenryck,P.; Saraswat,V and Deville,Y. 1995. Design, Implementation and Evaluation of the Constraint Language CC(FD), In Constraint Programming: Basics and Trends, A. Podelski Ed., Springer-Verlag.

[Kim & al 2001] Kim,P.; Williams, B adn Abramson, M. 2001. Executing Reactive, Model-based Programs through Graph-based Temporal Planning. Proceedings of the International Joint Conferent on Artificial Intelligence, Seattle, WA.