Resource reasoning in temporal planning

Filip Dvořák

Obsah

- Úvod
  - Plánování s časem a zdroji
  - Motivace
- Práce s časem
  - algebry, CSP, STP, DTP, exekuce
- Práce se zdroji
  - Taxonomie
  - Unární
  - Diskrétní
  - Rezervoáry

Plánování s časem a zdroji

- V modelu světa se snažíme najít plán, který z počátečního stavu světa dosáhne cíle.
- Plánování s explicitním časem.
  - Akce mají dobu trvání, jsou rozvrženy v čase a jejich efekty mohou interferovat.
- Proč zdroje v plánování?
  - Přenesením zdrojů do plánování můžeme sledovat kritéria zajímavá pro rozvrhování již během konstrukce plánu.

Motivace

- Autonomní systémy
  - Trex, Mars Rover, MrSPOCK
  - Armáda – letadla, družice
- Efektivita
  - Rescue operation
  - Manufacturing
  - Logistika
- „Strong AI“
  - Být schopen naplánovat cestu k dosažení svých cílů je klíčové.

Příklady

- **Manufacturing:**
  - *A grinding operation requires a grinding machine. The* ***grinding machine*** *can perform only one operation at a time. Only one grinding machine is available in the shop.*
- **Project Management:**
  - *In a web site development project, the task of testing browsers compatibility requires an effort of 6 person-months of a* ***quality engineer****. 5 quality engineers are available.*
- **Satellite:**
  - *When the satellite is not within range of a receiving station, it can store images on an onboard* ***solid-state memory*** *that provides a capacity of 90 Gbits (550 images).*
- **Crisis Management:**
  - *An aircraft B737-200 has a* ***capacity*** *to evacuate at most 120 people. Maximum* ***cargo loading*** *is 3,400kg for a* ***volume*** *of 14.3* m^3*.* ***Maximal range*** *of the aircraft is 2480km.*

Čas na čaj

- Zajímá nás konzistence a řešení.
- Kvalitativně
  - Point Algebra
    - A {<, =, >}B
      - Řešení v polynomiálním čase.
  - Interval Algebra
    - 13 vztahů mezi intervaly.
      - NP-těžké.
- Kvantitativně
  - STP, DTP

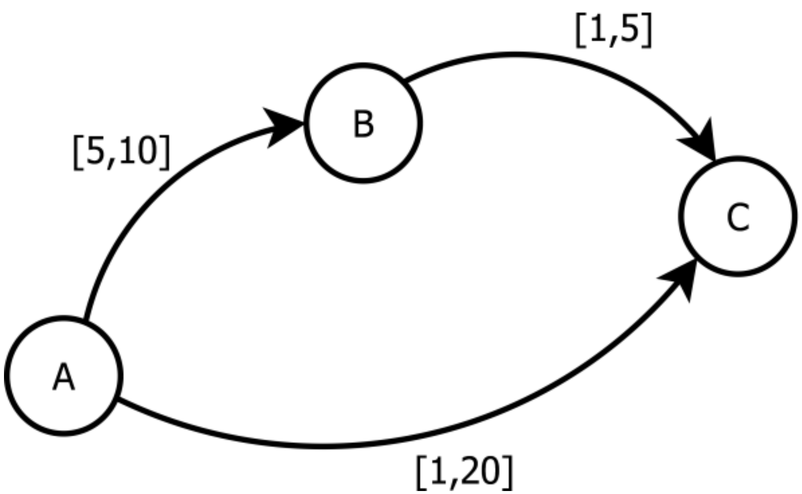
Constraint Satisfaction Problem

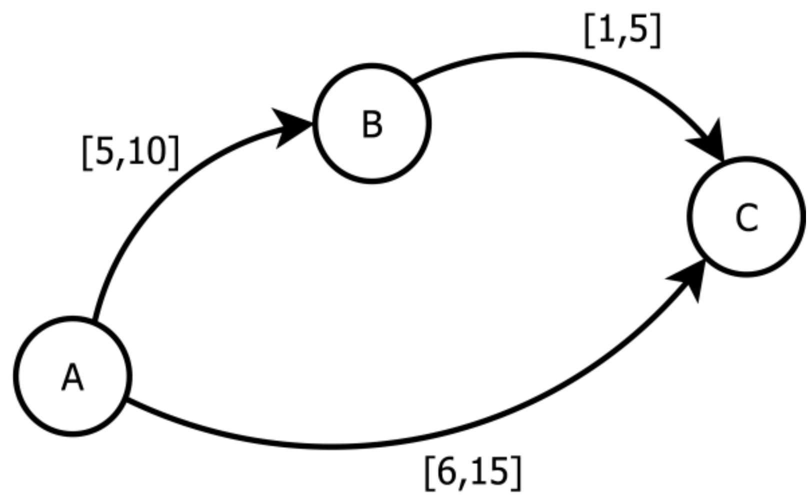
- Definice:
  - Proměnné: $X = \{x_1,\dots,x_n\}$
  - Domény: $D = \{d_1,\dots,d_n\}$
  - Podmínky: $C = \{c_1,\dots,c_m\}$
- (Instance) řešení:
  - Přiřazení hodnot proměnným za splnění podmínek.
- Způsob řešení:
  - Prohledávání, propagace podmínek.
  - NP-úplné !

Simple Temporal Problem

- Množina časových proměnných, reprezentujících události.
  - {A,B,C,...}
- Množina binárních podmínek
  - $x \leq B - A \leq y$, neboli B se stane nejdříve za x a nejpozději za y časových jednotek po A.
- Proměnné a podmínky tvoří časovou síť (STN)
  - Síť lze minimalizovat a odvodit tak mezní podmínky mezi všemi páry proměnných.
    - $O(n^3)$

Simple Temporal Network

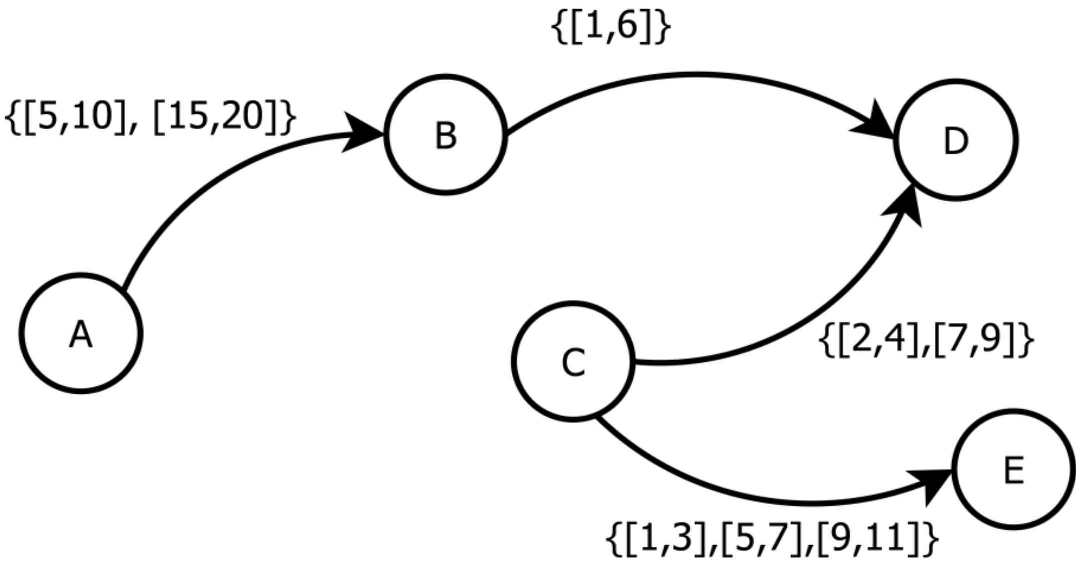




Disjunctive Temporal Problem

- Množina časových proměnných, reprezentujících události.
  - {A,B,C,…}
- Množina binárních podmínek
  - Každá podmínka je tvořena disjunkcí podmínek $(x_1 \leq B - A \leq y_1),\ldots,(x_k \leq B - A \leq y_k)$
    - NP-těžké už pro binární disjunkty.

Disjunctive Temporal Network



Exekuce

- Naivní modely
  - STP & DTP dispatch
- Kontrolovatelnost
  - Silná, slabá, dynamická

Taxonomie zdrojů

- Based on the way a resource is consumed and produced, we distinguish between resources that are:
  - *Consumable*
    - when the resource is only consumed in the system; e.g. fuel in a car, which cannot be refuelled.
  - *Producible*
    - when the resource is only produced in the system; e.g. some waste-product of industrial system.
  - *Replenishable*
    - when the resource can be both consumed and produced in the system; e.g. fuel in a car, which can be refuelled.
  - *Reusable*
    - when production and consumption must happen in tandem, e.g. for each consumption there exists a production.

Taxonomie

- Based on quantities that can be consumed or produced by a resource we distinguish between resources that are:
  - *Discrete*
    - when the resource is consumed, produced, or used in discrete quantities; e.g. sitting rooms in a car.
  - *Continuous*
    - when the resource is consumed, produced, or used in continuous quantities; e.g. fuel in a car.

Taxonomie

- Based on properties of capacity of a resource, we distinguish between:
  - *Single-capacity*
    - when the resource can be thought of as one unit, which must be consumed as a whole.
  - *Multi-capacity*
    - when the resource represents multiple units which can be used or consumed by different operations.
  - *Fixed Capacity*
    - when the capacity does not change over time.
  - *Variable Capacity*
    - when the capacity of the resource is a function of time; e.g. a battery whose capacity degrades.

Taxonomie

- Additionally we distinguish between resources that are:
  - *Shared*
    - when multiple activities can access the resource.
  - *Exclusive*
    - when only a single activity can access the resource.
  - *Single-dimensional*
    - when only a single level of the resource is considered; e.g. the number of places in an elevator.
  - *Multi-dimensional*
    - when multiple levels of the resource are considered; e.g. an elevator with the maximal allowed number of passengers and the maximal allowed weight.

Zdroje

- Unary resource:
  - Resource of capacity 1. Two activities requiring the same unary resource cannot overlap
- Discrete resource:
  - Resource of capacity Q. Activities requiring the same discrete resource can overlap provided the resource capacity Q is not exceeded
- Reservoir:
  - Resource of capacity Q and initial level L; Activities may consume or produce the reservoir. The level of the reservoir over time must be kept in the interval [0,Q]

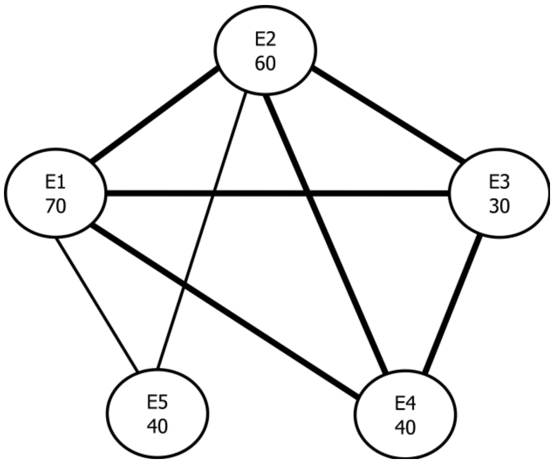
Techniky

- Disjunctive constraints
  - *end(A) < start(B) OR start(B) > end(A)*
- Edge-finding
  - detects situations where a given activity *A cannot execute after any activity in a set Ω* because there would not be enough time to execute all the activities in *Ω ∪ A between* the earliest start time of activities in *Ω ∪ A and the latest end time of activities in Ω ∪ A*. When such a situation occurs, it means that *A must execute before all the activities in Ω* and it allows computing a new valid upper bound for the end time of *A*.
- Energy constraint
  - A typical example of energetic reasoning consists in finding pairs of activities *(A,B) on* a unary resource such that ordering activity *A before B would lead to a dead end because* the unary resource would not provide enough "energy" between the earliest start time of *A and the latest end time of B to execute A, B and all the other activities that necessarily* need to partly execute on this time window.
- Energy precedence constraint
- Balance constraint

Precedence graph

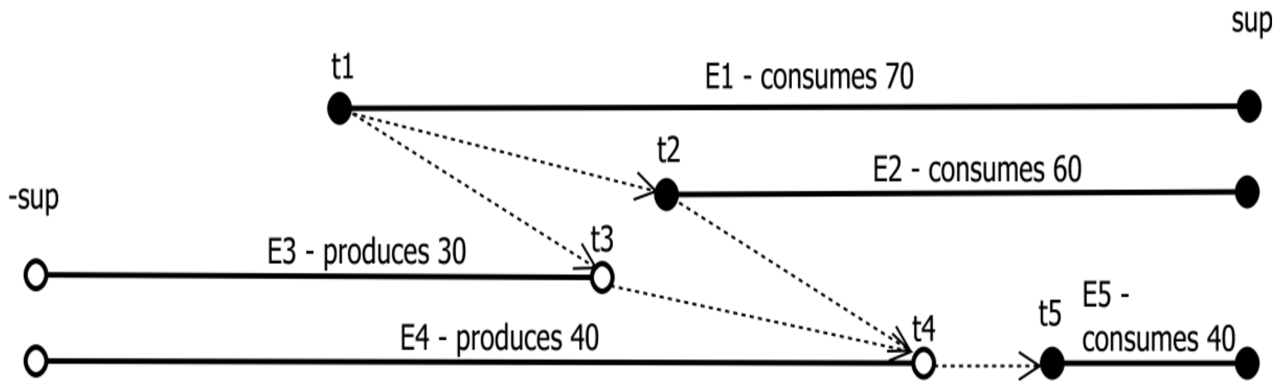
- Nodes correspond to requirements
- Edge between two nodes exits iff the time intervals of requirements corresponding to nodes *may* overlap.

Precedence graph



The precedence graph for a multi-capacity reusable resource. The capacity of the resource is 170 (capacity of the reservoir + requirements of all production events). Lines represent potential intersection of the time intervals of the requirements. Bold lines represent the complete subgraph that overconsumes the resource (70+40+30+60 > 170).

Transformace na diskrétní zdroj



- Transformation of the reservoir into a multi-capacity reusable resource. Dotted lines represent temporal relation *necessarily before*.

Zdroje

- Temporal and resource reasoning for planning, scheduling and execution, Nicola Muscettola, Martha Pollack, ICAPS2005
- Algorithms for propagating resource constraints in AI planning and scheduling: Existing approaches and new results, Philippe Laborie, Artificial Intelligence, 2003
- Constraint-based scheduling and planning, P. Baptiste, P Laborie, Hand of Constraint Programming, 2006
- Integrating Time and Resources into Planning, F. Dvořák, R. Barták, ICTAI 2010