The Relation between Vehicle Scheduling and Path Coloring

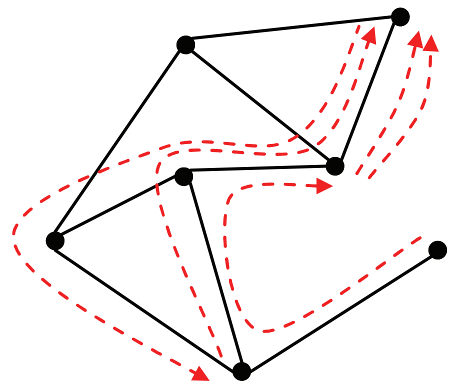
Evangelos Bampas, Georgia Kaouri, Michael Lampis, Aris Pagourtzis

National Technical University of Athens

Introduction

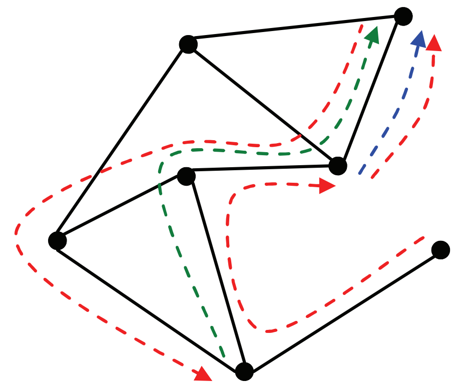
- VEHICLE SCHEDULING: A periodic scheduling problem with an objective function that seeks to maximize safety.
- PATH COLORING: A widely studied optimization problem with many known results.
- We present algorithms and hardness results for VS based on results for PC.
- Network topologies studied:
  - Chains
  - Rings
  - Spiders
  - Trees

Vehicle Scheduling



- Maximize the minimum time distance between successive trains that pass from the same edge in the same direction.
- This results in a more delay-tolerant system.

Path Coloring



- Assign different colors to paths sharing a directed edge.
- Minimize the number of colors used.
- Our aim is to construct a schedule from a coloring.

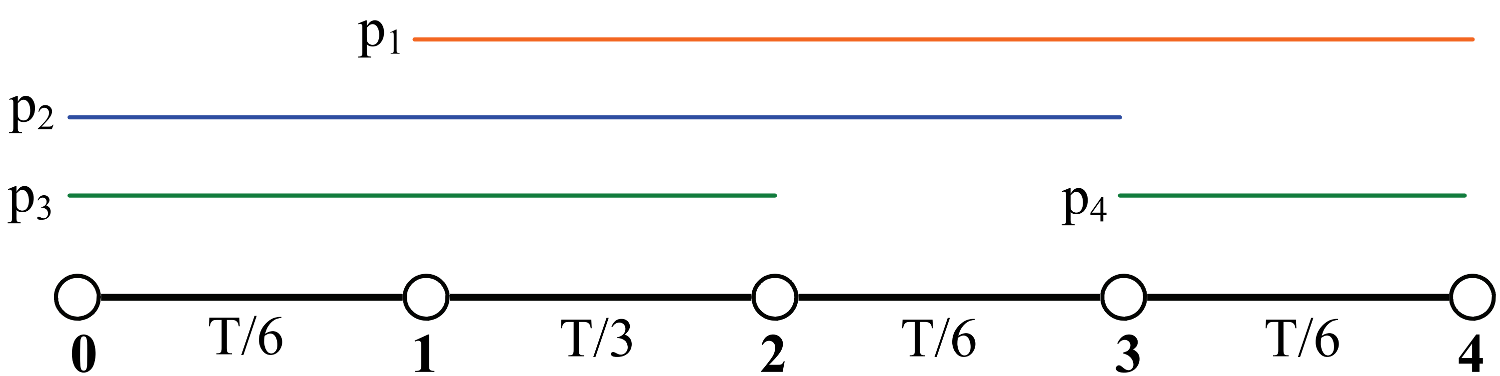
VS in a nutshell

- We are given a railway network graph with edges representing directed railway lines.
- All trains move at the same speed. The travel time for each edge is given.
- We are given a set of simple paths representing train routes.
- All routes are re-executed periodically (e.g. every day). The period T is given.
- The waiting time at stations is negligible.

Chains

- PC can be solved with L colors, where L is the maximum load on any edge.
- Divide the period into L timeslots: $0, \frac{T}{L}, 2\frac{T}{L}, \ldots, (L-1)\frac{T}{L}$. Assign the i-th timeslot to paths colored with the i-th color.
- For each path set starting time $s\text{–}time = timeslot + (\text{starting dist from } 0)$.
- Time distance $\geq \frac{T}{L}$ at node 0 and therefore everywhere, clearly optimal.

An example for chains

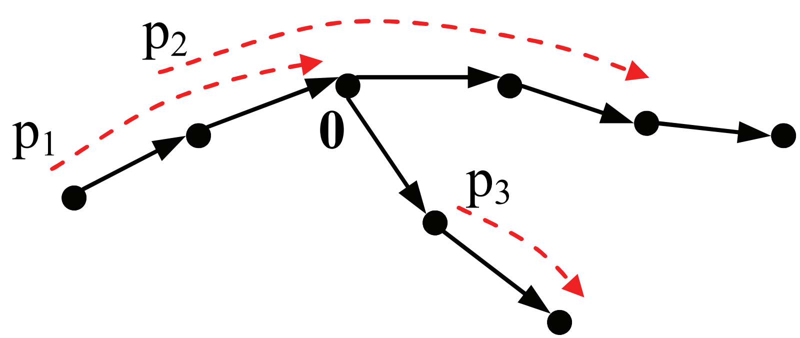


$$
\begin{aligned}
s\text{–}time(p_1) &= \frac{5T}{6} \\
s\text{–}time(p_2) &= \frac{T}{3} \\
s\text{–}time(p_3) &= 0 \\
s\text{–}time(p_4) &= \frac{2T}{3}
\end{aligned}
$$

Stars and Spiders

- Solve the corresponding PC (L colors) and assign timeslots.
- Compute starting times:
  - For paths passing through 0, or directed towards it set $s\text{–}time = timeslot - (\text{starting dist from } 0)$
  - For paths directed away from 0 set

    $s\text{–}time = timeslot + (\text{starting dist from } 0)$



Rings with perimeter $C \equiv_T 0$

- PC is *NP*-hard [Garey,Johnson,Miller,Papadimitriou].
- Given a path coloring with k colors, define k timeslots: $0, \frac{T}{k}, \ldots, (k-1)\frac{T}{k}$. Assign the i-th timeslot to paths colored with the i-th color.
- For each path set starting time $s\text{–}time = timeslot + (\text{starting dist from } 0)$.
- Time distance $\geq \frac{T}{k}$. How close is it to the optimal?

Reducing PC to VS in rings

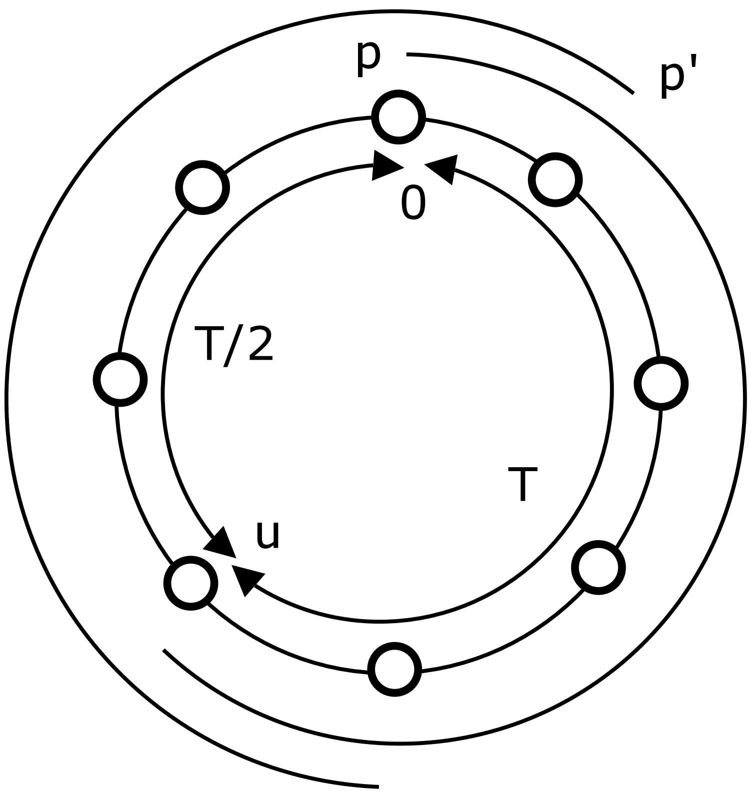
- Given a PC instance with n nodes, construct a VS instance with time distances 1 and $T = n$
- A coloring with k colors yields a schedule with distance $\frac{T}{k}$ by the previous algorithm.
- A schedule with distance $\frac{T}{k}$ yields a k-coloring by folowing the reverse procedure, i.e. assigning colors according to $s\text{–}time - (\text{starting dist from } 0)$.
- VS in rings is *NP*-hard

Rings $C \equiv_T 0$

- $\frac{T}{\mathrm{OPT}_{\mathrm{PC}}-1}$ is an upper bound to $\mathrm{OPT}_{\mathrm{VS}}$ by the previous reduction.
- A ρ-approximation for PC yields a $\frac{1}{\rho}\frac{L}{L+1}$ approximation for VS.
- Corollary: There is a $\frac{2}{3}\frac{L}{L+1}$ approximation algorithm for VS in rings where $C \equiv_T 0$ [Karapetian]

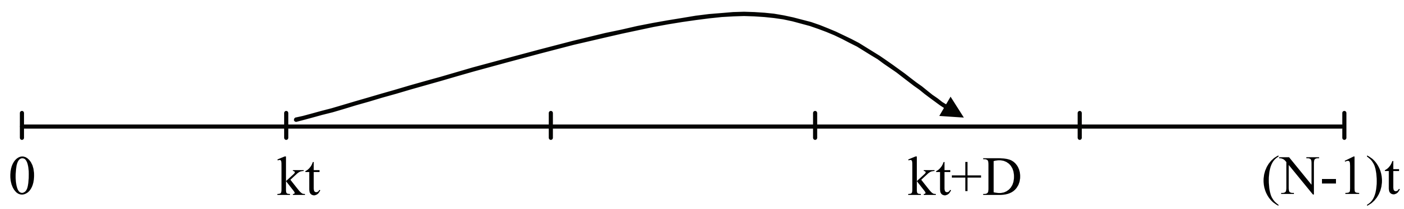
Rings ($C \not\equiv_T 0$)

- When $C \not\equiv_T 0$, $\frac{T}{\mathrm{OPT}_{\mathrm{PC}}}$ may not be achievable.



- This instance is colorable with 2 colors.
- A time distance of $\frac{T}{2}$ is impossible to achieve.

Rings ($C \not\equiv_T 0$)



- As a result we need more timeslots.
- When a timeslot kt is assigned, timeslots close to $kt + D$ are excluded, where $D = C \mod T$.

Rings ($C \not\equiv_T 0$)

- Split the paths into two sets $\mathcal{P}_0$ and $\mathcal{P}_c$.
- Solve the chain in $\mathcal{P}_c$ optimally.
- Enumerate paths in $\mathcal{P}_0$ according to their ending points.
- Assign timeslots to those paths.
- Set $s\text{–}time$ for $p \in \mathcal{P}_0$ s.t. p arrives at 0 at time $timeslot$.

Rings ($C \not\equiv_T 0$)

- We need $6L'$ timeslots, where $L' = \max\{L_c, L_0\}$.
- $\frac{T}{L'}$ is an upper bound on the optimal solution.
- We have achieved a time distance of $\frac{T}{6L'}$.
- As a result, a $\frac{1}{6}$- approximation algorithm is obtained.

Trees

- We study trees where the time distance for every edge is a multiple of $\frac{T}{2}$.
- In this case for the time distances τ between any three nodes a, b, c we have $\tau(a,b) + \tau(b,c) \equiv_T \tau(a,c)$.
- This property holds for rings where $C \equiv_T 0$ and is the only precondition to our results.

Trees $(T(e) \equiv_{\frac{T}{2}} 0)$

- Pick a node arbitrarily and label it 0.
- Given a path coloring with k colors, define k timeslots: $0, \frac{T}{k}, \ldots, (k-1)\frac{T}{k}$. Assign the i-th timeslot to paths colored with the i-th color.
- For each path set starting time $s\text{--}time = timeslot + (\text{starting dist from } 0)$.

Trees ($T(e) \equiv_{\frac{T}{2}} 0$)

- Suppose paths $p : a \to b$ and $p' : c \to d$ overlap and u is their first common node.
- p arrives at u at $timeslot(p) + \tau(0, a) + \tau(a, u) = timeslot(p) + \tau(0, u)$.
- p' arrives at u at $timeslot(p') + \tau(0, c) + \tau(c, u) = timeslot(p') + \tau(0, u)$.

Trees ($T(e) \equiv_{\frac{T}{2}} 0$)

- VS in trees is *NP*-hard.
- Given a ρ-approximation for PC in trees, a $\frac{1}{\rho}\frac{L}{L+1}$-approximation can be achieved for VS in this class of trees.
- Corollary: There is a $\frac{3}{5}\frac{L}{L+1}$-approximation algorithm for VS in this class of trees [Erlebach et al.].