Section 24.3: Feasibility Checking

Encoding as Satisfiability

Input: undirected graph $G=(V,E)$, # of colors k.

Decision variables: x_{vi} (for each $v \in V$, $i \in \{1,2,\ldots,k\}$).

[intended semantics: x_{vi} = TRUE $\Longleftrightarrow$ v assigned color i]

Constraints:

$\neg x_{ui} \vee \neg x_{vi}$ (for each $(u,v) \in E$, $i \in \{1,2,\ldots,k\}$) 1

$x_{v1} \vee x_{v2} \vee \cdots \vee x_{vk}$ (for each $v \in V$) 2

$\neg x_{vi} \vee \neg x_{vj}$ (for each $v \in V$, distinct $i,j \in \{1,2,\ldots,k\}$) 3

optional

Accommodating Side Constraints

Twist #1: may or may not be OK to assign two overlapping stations to adjacent channels (e.g., 14 and 15).

Solution: Compile a list of forbidden channel assignments for each pair of stations.

$\Rightarrow$ easy to incorporate into SAT formulation

e.g., $\neg x_{u,14} \vee \neg x_{v,15}$ rules out assigning u to 14 and v to 15

Twist #2: not all stations eligible for all k channels.

Solution: omit variable x_{vi} whenever station v forbidden from channel i.

General principle: Often easier to incorporate idiosyncratic side constraints into MIP/SAT solvers than problem-specific algorithms.

The Repacking Problem

Known in advance: list V of stations, allowable channels C_v for each $v \in V$, allowable channel pairs P_{uv} for each station pair $u,v \in V$.

Input: A subset $S \subseteq V$ of stations.

Output: Assignment of each station v to a channel of C_v, so that pair of assignments to $u,v \in P_{uv}$ for all $u \& v$.
[or, correctly declare that no such assignment exists] i.e., is "unpackable"

Time budget: one minute or less!
[SAT formulation has over 10k variables, over 1M constraints!]

off-the-shelf solvers: needed $\geq$ 10 min

Trick #1: Presolvers

[k=2]

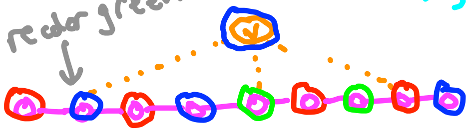
already not 2-colorable

Recall from quiz: every repacking instance has form $S \cup \{v\}$, where S is a packable set of stations.

① - Let T = neighbors of v in S. [stations with potential to interfere with v]

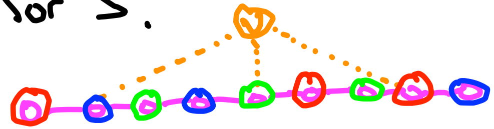
- If $T \cup \{v\}$ unpackable, return "unpackable."

[k=3]



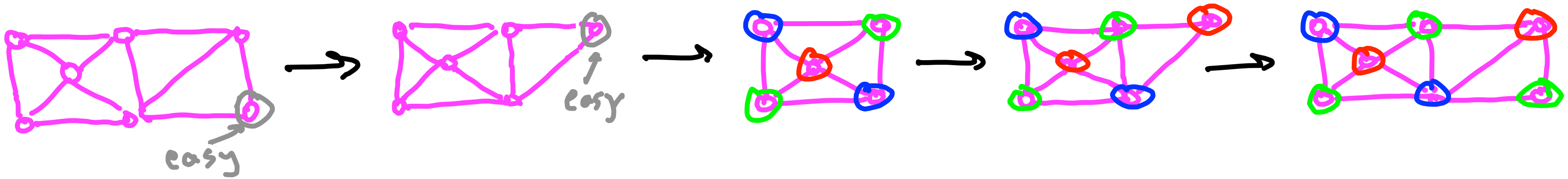
② - Inherit previous feasible channel assignments for S.

- Hold assignments of $S - T$ fixed.
- Assign channels to stations in $T \cup \{v\}$ so that combined assignments are feasible (if possible).



Trick #2: Preprocess and Simplify

[k=3]



Step 3: Iteratively prune easy stations (easy = no matter what neighbors' channel assignments are, ≥ one feasible channel left over for it)

[smaller instance is packable ⟺ original instance is]

Step 4: decompose into independent subproblems. (≈ connected components)

- if all packable, return union of channel assignments
- else return "unpackable"

[vertices = stations, edges = station pairs that could interfere]

Trick #3: A Portfolio of SAT Solvers

Idea: take advantage of:
(i) modern multi-core processors (8-core workstation)
(ii) heterogenity of running time across solvers + instances

Solution: use portfolio of 8 SAT solvers (run in parallel).

10k+ variables
1m+ constraints!

Result: solved >99% of repacking instances in ≤1 minute.

Question: what to do other 1% of the time?

Answer: if feasibility-checker times out on $S \cup \{v\}$, FCCGreedy assumes its unpackable. (can't risk violating feasibility)